

PAULO FABIANO URNAU GOTARDO

**SEGMENTAÇÃO DE IMAGENS DE PROFUNDIDADE
UTILIZANDO CURVATURAS DE SUPERFÍCIES E
UM MÉTODO DE ESTIMATIVA ROBUSTO**

Dissertação apresentada como requisito
parcial à obtenção do grau de Mestre.
Programa de Pós-Graduação em
Informática, Setor de Ciências Exatas,
Universidade Federal do Paraná.

Orientadora: Prof.^a Dr.^a Olga R. P. Bellon

CURITIBA

2002

PAULO FABIANO URNAU GOTARDO

**SEGMENTAÇÃO DE IMAGENS DE PROFUNDIDADE
UTILIZANDO CURVATURAS DE SUPERFÍCIES E
UM MÉTODO DE ESTIMATIVA ROBUSTO**

Dissertação aprovada como requisito parcial à obtenção do grau de Mestre no Programa de Pós-Graduação em Informática da Universidade Federal do Paraná, pela Comissão formada pelos professores:

Orientadora: Prof^a. Dr^a. Olga R. P. Bellon
Departamento de Informática, UFPR

Prof. Dr. Roberto Marcondes Cesar Jr.
Instituto de Matemática e Estatística, USP

Prof. Dr. André Luiz Pires Guedes
Departamento de Informática, UFPR

Curitiba, 14 de outubro de 2002

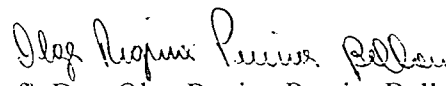


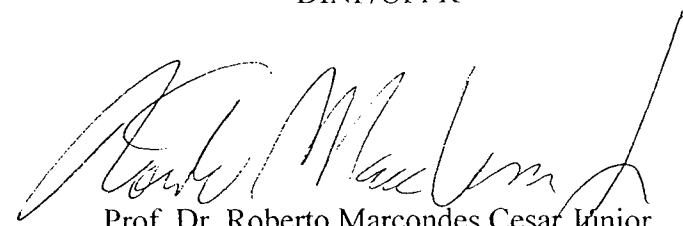
Ministério da Educação
Universidade Federal do Paraná
Mestrado em Informática


PARECER

Nós, abaixo assinados, membros da Banca Examinadora da defesa de Dissertação de Mestrado em Informática, do aluno *Paulo Fabiano Urnau Gotardo*, avaliamos o trabalho intitulado, "*Segmentação de Imagens de Profundidade utilizando Curvaturas de Superfícies e um Método de Estimativa Robusto*", cuja defesa foi realizada no dia 14 de outubro de 2002, às treze e trinta horas, no anfiteatro A do Setor de Ciências Exatas da Universidade Federal do Paraná. Após a avaliação, decidimos pela aprovação do candidato.

Curitiba, 14 de outubro de 2002.


Prof.^ª Dra. Olga Regina Pereira Bellon
DINF/UFPR


Prof. Dr. Roberto Marcondes Cesar Junior
IME/USP - Membro Externo


Prof. Dr. André Luiz Pires Guedes
DINF/UFPR



“... Se um dia, já homem feito e respeitado, sentires que a terra cede aos teus pés, que tuas obras se desmoronaram, que não há ninguém à tua volta para te estender a mão, esquece a tua maturidade, passa pela tua mocidade, volta à tua infância e balbucia, entre lágrimas e esperanças, as últimas palavras que sempre te restarão na alma: – Meu pai, minha mãe...”

Rui Barbosa

Aos meus pais, Paulo e Lenir.

Agradecimentos

Primeiramente, agradeço a Deus por toda a força, luz e proteção que me foram destinadas ao longo do meu caminho até aqui.

Agradeço aos meus pais e familiares pelos primeiros ensinamentos da minha vida, pelo amor a mim dedicado e pelo apoio nos momentos difíceis. Agradeço também à minha namorada, Ana Paula, pela compreensão e carinho. Amo muito todos vocês.

Agradeço muito à minha amiga professora Olga Bellon, pela orientação, motivação e pela confiança no meu potencial. Agradeço ao amigo Luciano Silva por toda a ajuda e atenção em nossas várias conversas, as quais certamente contribuíram muito para melhorar a qualidade deste trabalho.

Também dedico um agradecimento especial aos professores Alexandre Direne, André Pires Guedes, Hélio Pedrini e Marcos Sunye, pelo auxílio durante o mestrado e também nos preparativos para o doutorado. Agradeço ao professor Roberto Marcondes Cesar Jr. pelos importantes comentários e sugestões durante a defesa desta dissertação.

Agradeço aos meus colegas de mestrado Éverton Vieira e Maurício Severich pelo companheirismo e por todas as coisas que tornaram mais divertidos estes 20 meses de trabalho no IMAGO.

Por fim, a todos os amigos, dentro e fora da Universidade Federal do Paraná, que contribuíram direta ou indiretamente para a realização deste trabalho, aqui vai meu muito obrigado!

P.F.U.G.

Resumo

Neste trabalho são apresentadas contribuições para o aperfeiçoamento da segmentação de imagens de profundidade, um processo de fundamental importância a sistemas de visão computacional e ainda um dos maiores desafios nesta área de pesquisa. O principal objetivo é desenvolver técnicas de segmentação que preservem melhor a topologia dos objetos em cena, de modo a auxiliar processos posteriores de representação, modelagem, reconhecimento e reconstrução de objetos, ajudando a diminuir algumas limitações na utilização de sistemas de visão computacional.

O problema da segmentação de imagens de profundidade foi abordado em duas formas diferentes e as principais contribuições apresentadas são: (1) dois métodos de detecção de bordas inéditos baseados em valores das curvaturas de superfície H e K e integrando dados de profundidade e de intensidade luminosa correspondentes à mesma cena; e (2) um novo método de segmentação de imagens de profundidade utilizando um algoritmo genético e um método de estimativa robusto, aperfeiçoados, para a extração de superfícies planas das imagens.

Através da utilização de uma mesma base de imagens, os resultados experimentais foram comparados positivamente aos resultados obtidos por outros quatro métodos de segmentação de imagens de profundidade, considerados a principal referência no assunto de acordo com a literatura. Os métodos de detecção de bordas integrando dados de intensidade luminosa preservam melhor as formas e localizações de bordas dos objetos em cena e podem ser utilizados para melhorar os resultados obtidos por outros métodos de segmentação. O método de segmentação por extração de superfícies planas foi avaliado quantitativamente, utilizando um conjunto de métricas relacionadas a segmentações manualmente geradas, e apresentou um melhor desempenho na preservação da topologia dos objetos, principalmente, pelo fato de melhor segmentar regiões pequenas das imagens.

As contribuições apresentadas constituem avanços relevantes para o aperfeiçoamento da segmentação de imagens de profundidade e estão sendo utilizadas como suporte a um projeto mais amplo, o SRIC3D, em desenvolvimento pelo grupo de pesquisa IMAGO.

Abstract

This work is a contribution to the improvement of the range image segmentation process, which is of fundamental importance to computer vision systems and still one of the greatest challenges in this research field. The main objective is to develop segmentation techniques to better preserve the objects topology in the imaged scenes in order to support object representation, modeling, recognition and reconstruction processes, while also helping to make feasible new applications of computer vision systems.

The range image segmentation problem was approached in two different ways and the main contributions presented here are: (1) two original edge detection techniques based on H and K surface curvature values and integrating range and light intensity data corresponding to the same scene; and (2) a novel range image segmentation method employing an improved genetic algorithm and a robust estimator to extract planar surfaces from the range images.

By the use of a same image database, the experimental results were positively compared to the ones obtained by other four range image segmentation methods which are considered the main reference in this subject, according to the literature. The edge detection techniques, integrating range and intensity data, better preserve shapes and edge locations of the imaged objects and may be applied to improve the performance of other segmentation methods. The segmentation method based on planar surface extraction was quantitatively evaluated, using a set of metrics related to ground truth segmentations, and presented a better performance in preserving object topology, mainly, because of the better segmentation of small image regions.

The presented contributions are relevant advances to the improvement of the range image segmentation process and are already being used as support for another project – the *Content-based Image Retrieval System of 3D Digital Replicas from Physical Objects*, SRIC3D – under development by the IMAGO research group.

Sumário

Lista de Figuras	viii
Lista de Tabelas	xi
1 Introdução	1
1.1 Sistemas de Visão Computacional	1
1.1.1 Estágios de Processamento	2
1.1.1.1 Aquisição	2
1.1.1.2 Melhoramento	3
1.1.1.3 Segmentação	3
1.1.1.4 Reconhecimento	4
1.2 Problemas na Segmentação de Imagens	4
1.3 Objetivos	6
1.4 Contribuições	7
1.5 Organização	8
2 Segmentação de Imagens	10
2.1 Imagens Digitais	10
2.1.1 Imagens de Intensidade Luminosa	10
2.1.2 Imagens de Profundidade	11
2.2 Integração de Imagens	13
2.3 Segmentação de Superfícies Digitais	14
2.3.1 Definição do Problema	14
2.3.2 Abordagens	17
2.3.2.1 Métodos de Agrupamento	17
2.3.2.2 Crescimento de Regiões	17
2.3.2.3 Divisão-e-União	18
2.3.2.4 Detecção de Bordas	18
2.3.2.5 Integrando Bordas e Regiões	19
2.4 Aproximação de Superfícies	19
2.5 Comparação de Resultados	21
2.5.1 Método de Avaliação	22

2.5.2	Algoritmos Comparados	24
2.5.2.1	University of South Florida (USF), EUA	25
2.5.2.2	Washington State University (WSU), EUA	25
2.5.2.3	University of Bern (UB), Suíça	25
2.5.2.4	University of Edinburgh (UE), Escócia	26
3	Segmentação Baseada em Curvaturas de Superfícies	27
3.1	Curvaturas de Superfícies	28
3.1.1	Cálculo das Curvaturas	30
3.2	Trabalhos Relacionados	32
3.2.1	Segmentação Baseada em Regiões	32
3.2.1.1	Besl e Jain	32
3.2.1.2	Yokoya e Levine	33
3.2.1.3	Trucco e Fisher	34
3.2.2	Segmentação por Detecção de Bordas (Fan <i>et al.</i>)	34
3.3	Métodos Desenvolvidos	35
3.3.1	Detecção de Bordas em Mapas de Sinais HK	37
3.3.1.1	Suavização	37
3.3.1.2	Cálculo das curvaturas H e K	39
3.3.1.3	Cálculo do mapa de sinais HK	40
3.3.1.4	Refinamento do mapa de sinais	42
3.3.1.5	Detecção de bordas	43
3.3.1.6	Integração dos mapas de bordas	44
3.3.2	Detecção de Bordas Utilizando a Curvatura H	45
3.3.2.1	Detecção de bordas	46
3.3.3	Resultados Experimentais	47
3.4	Conclusão	59
4	Segmentação por Extração de Superfícies	62
4.1	<i>Outliers</i> e o Método <i>Least Squares</i>	63
4.2	Métodos de Estimativa Robustos	64
4.2.1	<i>Least Median of Squares</i>	65
4.2.2	<i>Least Trimmed Squares</i>	68
4.2.3	<i>Random Sample Consensus</i>	68
4.2.4	<i>M-estimator Sample Consensus</i>	70
4.2.5	Discussão	71
4.3	Extração de Primitivas Geométricas	72
4.3.1	Algoritmos Genéticos	73
4.3.1.1	Descrição	74
4.3.1.2	Aplicação na Extração de Primitivas Geométricas	76

4.4	Trabalhos Relacionados	77
4.4.1	Otimização por Amostragem Aleatória	77
4.4.1.1	Roth e Levine	77
4.4.1.2	Lee, Meer e Park	79
4.4.2	Otimização por Algoritmo Genético	81
4.4.2.1	Roth e Levine	81
4.4.2.2	Chen e Liu	83
4.4.2.3	Yu, Bui, e Krzyzak	86
4.5	Método Desenvolvido	87
4.5.1	Pré-processamento	88
4.5.2	Extração de Superfícies	90
4.5.2.1	Particionamento do Espaço de Busca	91
4.5.2.2	Aproximação de Superfície	92
4.5.2.3	Crescimento e Extração da Região	96
4.5.2.4	Atualização do Espaço de Busca	98
4.5.3	Resultados Experimentais	99
4.5.4	Avaliação Quantitativa	117
4.6	Conclusão	121
5	Conclusão e Trabalhos Futuros	123
	Referências Bibliográficas	126

Lista de Figuras

1.1	Relacionamento entre áreas de pesquisa	2
1.2	Segmentações geradas manualmente e por métodos computacionais.	6
2.1	Imagem de intensidade luminosa utilizada em projetos de futebol de robôs.	11
2.2	Dois tipos de imagens para uma mesma cena.	12
2.3	Perfis de bordas.	18
2.4	Ruído impulsivo e de quantização.	20
3.1	Superfícies classificadas pelos sinais das curvaturas H e K.	29
3.2	Comportamento dos valores de curvatura.	35
3.3	Exemplo de imagens obtidas com um sensor Perceptron (perc.test.26)	36
3.4	Diagrama de estágios de processamento do método de detecção de bordas. . .	37
3.5	Efeito de um filtro de suavização Gaussiano nas bordas de uma imagem. . .	38
3.6	Aplicação de filtro mediano para eliminar ruído impulsivo nas imagens. . .	38
3.7	Imagens de curvaturas obtidas da imagem de intensidade luminosa.	40
3.8	Imagens de curvaturas obtidas da imagem de profundidade.	40
3.9	Mapas de sinais HK para as duas imagens de entrada.	41
3.10	Corte transversal de uma borda entre regiões <i>ridge</i> e <i>valley</i>	42
3.11	Mapas de sinais HK após reclassificação de regiões de ruído.	43
3.12	Mapas de sinais HK após dilatação de regiões <i>ridge</i> e <i>valley</i>	43
3.13	Mapas de bordas extraídos dos mapas de sinais HK.	44
3.14	Mapa de bordas resultante da integração.	45
3.15	Diagrama de estágios de processamento do método de detecção de bordas. . .	46
3.16	Deteção de bordas utilizando <i>zero-crossings</i> da curvatura H.	46
3.17	Mapas de bordas detectadas a partir dos valores de curvatura utilizando informação de magnitude de gradiente.	47
3.18	Mapas de bordas para imagem de testes Perceptron número 26.	49
3.19	Mapas de bordas para imagem de testes Perceptron número 0.	50
3.20	Mapas de bordas para imagem de testes Perceptron número 18.	51
3.21	Mapas de bordas para imagem de testes Perceptron número 20.	52
3.22	Mapas de bordas para imagem de testes Perceptron número 22.	53
3.23	Mapas de bordas para imagem de testes Perceptron número 25.	54
3.24	Mapas de bordas para imagem de testes Perceptron número 27.	55

3.25	Mapas de bordas para imagem de testes Perceptron número 28.	56
3.26	Mapas de bordas para imagem de testes Perceptron número 29.	57
4.1	Aproximações incorretas do método LS devido à presença de ruídos	63
4.2	Aproximações corrompidas por <i>outliers</i> e <i>pseudo-outliers</i>	64
4.3	Hipóteses geradas pelo LMS para aproximação de reta	66
4.4	Tolerância de erro (faixa de <i>inliers</i>) do RANSAC	69
4.5	Aproximação corrompida em descontinuidade de pequena magnitude . . .	71
4.6	Diagrama de execução de um algoritmo genético.	74
4.7	Operações genéticas de <i>crossover</i> e mutação.	76
4.8	Diferenças no comportamento das faixas de <i>inliers</i> do RANSAC e do MSAC. .	79
4.9	Operação de <i>crossover</i> uniforme.	82
4.10	Extração de superfície de extensão desconhecida.	85
4.11	Distâncias algébrica e geométrica entre um ponto e um plano.	87
4.12	Estágios de processamento do método de extração de superfícies.	88
4.13	Imagem de testes Perceptron número 29.	89
4.14	Segmentação rústica inicial para determinação de espaços de busca. . . .	91
4.15	Correta classificação de <i>pseudo-inliers</i> como <i>outliers</i>	93
4.16	Mutação por deslocamento do ponto dentro do espaço de busca.	96
4.17	Crescimento de superfície em região de borda de orientação.	98
4.18	Imagens do experimento de determinação dos parâmetros do AG.	101
4.19	Evolução média dos valores da função de custo do método robusto utilizando diferentes processos de otimização.	102
4.20	Regiões segmentadas para imagem de testes Perceptron número 0.	104
4.21	Regiões segmentadas para imagem de testes Perceptron número 18. . . .	105
4.22	Regiões segmentadas para imagem de testes Perceptron número 19. . . .	106
4.23	Regiões segmentadas para imagem de testes Perceptron número 20. . . .	107
4.24	Regiões segmentadas para imagem de testes Perceptron número 22. . . .	108
4.25	Regiões segmentadas para imagem de testes Perceptron número 23. . . .	109
4.26	Regiões segmentadas para imagem de testes Perceptron número 25. . . .	110
4.27	Regiões segmentadas para imagem de testes Perceptron número 26. . . .	111
4.28	Regiões segmentadas para imagem de testes Perceptron número 27. . . .	112
4.29	Regiões segmentadas para imagem de testes Perceptron número 28. . . .	113
4.30	Regiões segmentadas para imagem de testes Perceptron número 29. . . .	114
4.31	Identificação de regiões de imagens Perceptron correspondentes a objetos em cena a partir dos dois maiores planos da imagem e de bordas de profundidade.	116
4.32	Segmentação de superfícies curvas por aproximações planares.	117
4.33	Número médio de regiões corretas de acordo com nível de tolerância. . . .	118
4.34	Ocorrência de sobre-segmentação em função do nível de tolerância. . . .	119

4.35	Ocorrência de sub-segmentação em função do nível de tolerância.	119
4.36	Número médio de regiões perdidas de acordo com nível de tolerância. . . .	120
4.37	Número médio de regiões de ruído de acordo com nível de tolerância. . . .	120

Lista de Tabelas

3.1	Classificação de superfícies utilizando as curvaturas principais.	28
3.2	Classificação de superfícies utilizando as curvaturas H e K.	29
4.1	Número de hipóteses k para índice de confiança $P_s = 0,99$	67
4.2	Avaliação média para os métodos de segmentação aplicados às imagens de testes Perceptron à tolerância de 80%.	118

Capítulo 1

Introdução

Sistemas de percepção computacional têm como objetivo interpretar informações – na forma de sinais como som, luz ou distâncias – sobre o ambiente no qual está inserido e gerar uma ação de resposta útil para a aplicação em questão. Por exemplo, um sistema de reconhecimento de fala pode imprimir uma palavra ao reconhecê-la a partir dos sinais sonoros de entrada.

Uma das classes de sistemas de percepção é constituída pelos *sistemas de visão computacional*, os quais buscam reproduzir as capacidades do sistema de visão humano e agir de acordo com estímulos visuais. Estes sistemas têm como objetivo a interpretação de uma cena descrita por um conjunto de dados de entrada na forma de *imagens digitais*. Para entender as informações sobre o mundo real, tais sistemas devem localizar e reconhecer, de forma confiável, os objetos presentes na imagem. Neste processo de interpretação, a fase de *segmentação*, descrita na próxima seção, é de fundamental importância e melhorar o seu desempenho constitui o principal objetivo do trabalho aqui apresentado.

1.1 Sistemas de Visão Computacional

O número de aplicações para sistemas de visão computacional vem crescendo muito nos últimos anos, principalmente nos campos da indústria, biologia, medicina, cultura (e.g. arqueologia) e naqueles cujos processos demandem precisão, confiabilidade e tempo de resposta iguais ou melhores que os produzidos por seres humanos (e.g. auxílio a deficientes visuais, recuperação de imagens em bibliotecas digitais).

Na indústria, sistemas de visão computacional são utilizados para a inspeção automática de peças e circuitos impressos (controle de qualidade), monitoramento de processos, robótica de automação, navegação autônoma e engenharia reversa de protótipos. No campo de biologia e medicina, destacam-se os processos de biometria, seleção e contagem de células e as tarefas de identificação e reconhecimento de padrões para diagnóstico de doenças, os quais vêm sendo auxiliados cada vez mais por técnicas

de visão computacional. Outras aplicações são encontradas, por exemplo, nas áreas de sensoriamento remoto, processamento de documentos e vigilância.

Um sistema de visão computacional deve ser capaz de desempenhar suas funções da forma mais independente possível, necessitando de pouca, ou nenhuma, supervisão humana. Por isso, a área de pesquisa em visão computacional originou-se de uma ramificação da área de inteligência artificial, mas está relacionada a outras áreas como processamento de imagens, computação gráfica, geometria computacional, reconhecimento de padrões e psicofísica [Jain *et al.*, 1995]. Esta última tem contribuído muito para o desenvolvimento de modelos computacionais baseados no sistema de visão humano. A figura 1.1 ilustra o relacionamento entre processamento de imagens, computação gráfica (síntese de imagens) e a análise de imagens realizada em visão computacional. Resumidamente, computação gráfica pode ser entendida como a área que estuda a geração de representações visuais a partir de especificações geométricas; o processamento de imagens envolve técnicas de transformação de imagens visando melhorar suas características visuais; e a análise de imagens procura obter informações geométricas para os componentes de uma imagem, a partir de sua representação visual.

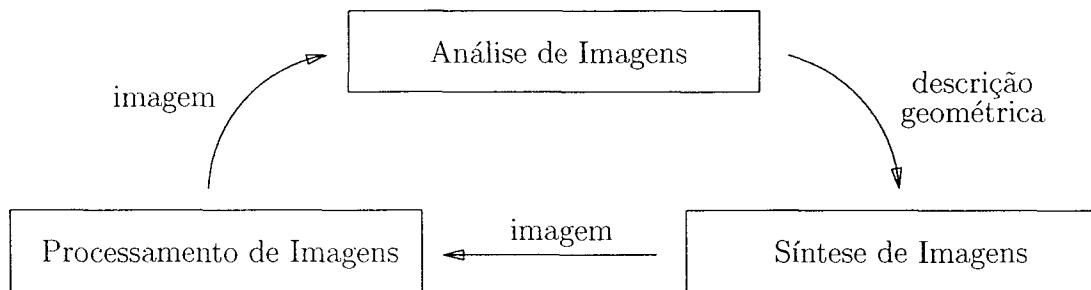


Figura 1.1: *Relacionamento entre áreas de pesquisa*

1.1.1 Estágios de Processamento

O processo de análise de uma cena do mundo real, representada em imagens digitais, realizado por sistemas de visão computacional pode ser decomposto em quatro principais estágios de processamento:

1.1.1.1 Aquisição

Inicialmente, as imagens de entrada são adquiridas através do uso de um dispositivo de captura, o sensor. O desempenho do sistema está fortemente relacionado à qualidade dos dados de entrada e também à sua natureza. O significado dos valores contidos nas imagens vai depender do tipo de dispositivo utilizado e vai influenciar na definição da abordagem mais adequada para interpretar estes dados.

Desde as primeiras pesquisas [Rosenfeld, 1969], a maioria dos sistemas de visão computacional tem utilizado como dados de entrada imagens de intensidade luminosa, que

contêm informações sobre a quantidade de luz refletida nos pontos das superfícies em cena. Imagens como estas são providas por sensores como câmeras de vídeo comuns, de baixo custo e tecnologia mais desenvolvida, oferecendo, portanto, dados mais confiáveis. Porém, uma grande limitação deste tipo de imagens é a ausência de informação tridimensional (3D) sobre a cena observada, o que dificulta sua interpretação.

Por outro lado, nas últimas duas décadas, técnicas de visão computacional vêm sendo extensivamente adaptadas e desenvolvidas de modo a serem aplicadas a imagens de profundidade (*range images*), geralmente obtidas com rastreadores a laser. Estas imagens contêm informações sobre as distâncias entre o sensor e os pontos nas superfícies dos objetos, apresentando, assim, informação 3D explícitas e úteis para a caracterização geométrica dos objetos em cena. Isso justifica a atenção que essas imagens vêm recebendo, apesar de os sensores que as produzem ainda serem caros e sua tecnologia relativamente recente e menos desenvolvida. Entretanto, imagens de profundidade apresentam também um maior nível de ruído, o que corresponde a degradações e imprecisões nos valores medidos pelo sensor.

1.1.1.2 Melhoria

Devido a interferências sofridas pelo sensor, freqüentemente as imagens apresentam degradações, também chamadas de ruídos. Estes ruídos introduzem efeitos indesejáveis nas etapas de processamento posteriores e, por isso, muitas vezes faz-se necessário, após a aquisição da imagem, realizar um processo de melhoria. Este processo utiliza técnicas de processamento de imagens para *filtrar* os dados contidos na imagem visando a diminuir o nível de ruído.

Entretanto, durante o processo de filtragem, distinguir os pontos da imagem que foram afetados por ruídos daqueles que não foram é uma tarefa difícil e, por isso, na grande maioria das vezes, o processo acaba produzindo efeitos colaterais, como o borramento das bordas e cantos dos objetos. A literatura descreve uma grande variedade de filtros [Trucco, 1992, Umasuthan e Wallace, 1996] desenvolvidos com o objetivo de minimizar estes efeitos indesejados, e a solução deste problema constitui uma área de grande interesse dentro do campo de visão computacional.

Além da filtragem de ruídos, a fase de melhoria também tem como finalidade realçar nas imagens algumas de suas características mais importantes, as quais serão utilizadas na seqüência, durante o processo de segmentação.

1.1.1.3 Segmentação

Uma etapa fundamental para a interpretação de imagens é o processo de segmentação. Como a interpretação de uma grande quantidade de dados por um sistema autônomo é um processo muito complexo, é necessário que esses dados sejam particionados (segmentados), formando estruturas de mais alto nível, que possam ser comparadas a modelos do mundo

real. Essas estruturas, ou *regiões*, em uma imagem são grupos de pontos com propriedades semelhantes. Regiões são importantes para a interpretação de uma imagem porque podem corresponder a objetos ou partes de objetos em uma cena. Logo, para uma imagem ser interpretada adequadamente, ela deve ser corretamente particionada em regiões que realmente correspondam aos objetos ou a partes dos objetos em cena. Um processo de segmentação que identifique corretamente as formas, topologia e localização das bordas dos objetos é um requisito de fundamental importância para que as informações inferidas pelo sistema de visão sejam confiáveis. Aqui, o termo *topologia de objetos* refere-se às conexões e ao posicionamento relativo entre as entidades geométricas (i.e. pontos, curvas e superfícies) que constituem os objetos.

No estágio de segmentação são extraídas características das imagens para que pontos que apresentem-nas semelhantemente possam ser agrupados, formando regiões distintas e homogêneas. Para isso, a escolha de quais características serão analisadas e a precisão do método utilizado para extrai-las são extremamente importantes. Esta escolha é totalmente dependente do tipo de sensor utilizado, do domínio da aplicação e de informações prévias sobre que tipo de objetos as cenas a serem analisadas podem conter.

Existem, basicamente, duas abordagens para a segmentação de imagens digitais. Uma delas é a *segmentação baseada em regiões* [Celinski, 1998], que procura gerar as regiões homogêneas da imagem através do agrupamento de pontos com características semelhantes. A outra abordagem é a *segmentação por detecção de bordas* [Silva, 2000], que, ao invés de analisar semelhanças, procura determinar linhas de disparidade, descontinuidades, para isolar as regiões homogêneas da imagem.

1.1.1.4 Reconhecimento

Esta etapa é fortemente dependente da etapa de segmentação, cujos resultados devem ser precisos para que os objetos em cena sejam identificados (reconhecidos) de forma confiável. No estágio de reconhecimento, representações das estruturas obtidas anteriormente são analisadas e comparadas a modelos do mundo real, previamente inseridos no sistema de visão. Tais modelos variam de acordo com o domínio de aplicação do sistema e podem restringir o seu uso para outros fins. Por isso, um sistema de visão genérico é aquele que se utiliza do menor número possível de informações prévias sobre o ambiente ao qual está inserido e realiza a maior parte das inferências baseando-se apenas nos dados contidos nas imagens de entrada.

1.2 Problemas na Segmentação de Imagens

Trabalhos de pesquisa realizados na área de visão computacional tem utilizado diferentes dispositivos e métodos computacionais, conforme o domínio de aplicação desejado. Entretanto, ainda não existem métodos ou dispositivos genéricos, que apresentem um

desempenho ótimo ou flexibilidade suficiente para serem empregados em várias situações sem necessitar de esforços de desenvolvimento adicionais. Existem vários problemas que fazem da análise de uma cena uma tarefa complexa, como por exemplo:

- Limitações dos sensores quanto à resolução das imagens e ao alto nível de ruídos;
- Variações na iluminação do ambiente e a existência de sombras e reflexos;
- A presença de diversos objetos com formas e tamanhos variados;
- Mudanças na visualização de um mesmo objeto devido a efeitos de translação, rotação, escala ou projeção;
- Oclusão, total ou parcial, de alguns objetos em cena devido à disposição destes em relação ao sensor;

Para que um grande número de aplicações sejam viabilizadas, sistemas de visão computacional devem saber lidar com vários desses problemas simultaneamente. Contudo, a segmentação de imagens, um estágio de fundamental importância, ainda é um dos maiores desafios no área de pesquisa em visão computacional. Apesar do grande número de métodos apresentados na literatura, o problema da segmentação permanece sem uma solução adequada, mesmo para cenas mais simples como aquelas contendo apenas objetos com superfícies planas (poliédricos) [Hoover *et al.*, 1996].

Processar uma imagem de modo a segmentar uma cena com diferentes tipos de objetos é uma tarefa difícil, principalmente, por ser extremamente dependente da correta extração de características dos objetos a partir de imagens ruidosas. Ruídos podem levar métodos de segmentação a distorcer as formas dos objetos, prejudicando seu reconhecimento: regiões distintas podem ser incorretamente identificadas como uma única região (*sub-segmentação*), ou uma região homogênea pode ser dividida em partes menores (*sobre-segmentação*).

Estes problemas são ilustrados na figura 1.2, onde a figura 1.2(a) contém a segmentação ideal para uma imagem, na forma de um mapa de bordas manualmente gerado, e as figuras 1.2(b) e 1.2(c) contém exemplos de segmentações produzidas por métodos computacionais. Nestes exemplos, a presença de regiões inexistentes e de bordas serrilhadas e incorretamente localizadas resultam na incorreta identificação das formas dos objetos em cena, atrapalhando o processo posterior de reconhecimento. Isso porque, no reconhecimento de objetos, são geradas representações (e.g. poligonais) das regiões segmentadas e essas representações são então comparadas a modelos geométricos. Formas incorretas de regiões causam erros nas representações e, conseqüentemente, no reconhecimento.

As características comumente utilizadas por métodos de segmentação de imagens são as características geométricas das superfícies dos objetos como, por exemplo, vetor gradiente, orientação e grau de curvatura. Assim, devido à importância da caracterização geométrica durante o processo de segmentação, imagens de profundidade têm sido cada

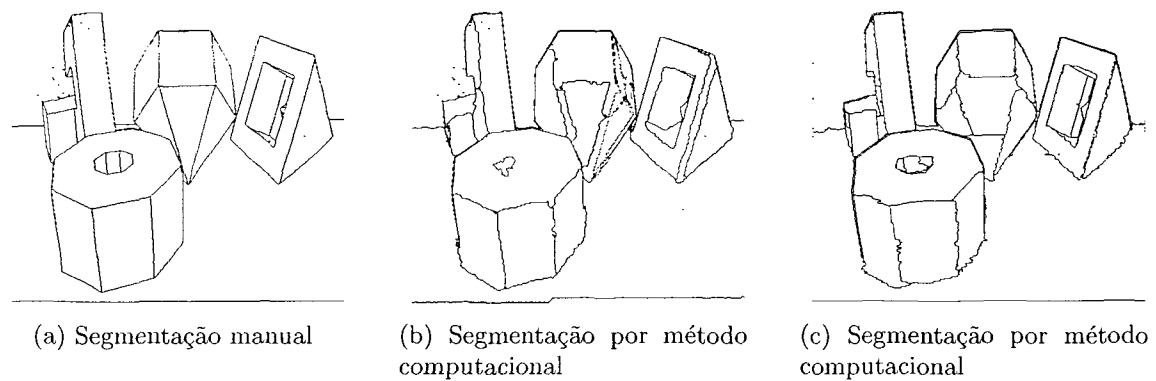


Figura 1.2: Segmentações geradas manualmente e por métodos computacionais.

vez mais utilizadas como dados de entrada por apresentarem informações tridimensionais explícitas.

Apesar disso, o cálculo de valores para as características geométricas requer a utilização de métodos de aproximação de superfícies, os quais precisam lidar com problemas como a escolha do modelo de superfície adequado e a interferência, durante a aproximação, de pontos correspondentes a ruído ou de pontos de outras superfícies, vizinhas. Informação prévia sobre os objetos em cena pode contribuir para a escolha do modelo, mas pode diminuir a generalidade do método. Apesar de as curvaturas serem muito úteis para a classificação de superfícies, ajudando na escolha do modelo, os métodos frequentemente utilizados para calculá-las são extremamente sensíveis a ruídos.

Nesse contexto, diferentes técnicas, como métodos de estimativa robustos [Stewart, 1999] [Meer *et al.*, 2000] e algoritmos genéticos [Man *et al.*, 1996], tem sido utilizadas nos últimos anos para se obter valores mais precisos para características de superfícies, na tentativa de solucionar o problema da segmentação.

1.3 Objetivos

Devido à sua importância e ao fato de ainda não existirem soluções definitivas para o problema de segmentação de imagens, o principal objetivo do trabalho aqui relatado é contribuir para a solução deste problema através do desenvolvimento de novas técnicas de segmentação de imagens de profundidade. Espera-se que essas técnicas preservem melhor a topologia dos objetos em cena, de modo a auxiliar processos posteriores de representação, modelagem, reconstrução e reconhecimento desses objetos, ajudando a diminuir algumas limitações na utilização de sistemas de visão computacional.

Na busca por soluções, o problema da segmentação de imagens de profundidade foi abordado em duas frentes diferentes [Gotardo *et al.*, 2002]:

Uma das abordagens consistiu na realização de experimentos em detecção de bordas a partir de curvaturas de superfícies para fins de segmentação. Nestes experimentos foram

utilizadas informações de curvaturas calculadas para pares de imagens contendo, cada um, uma imagem de intensidade luminosa e uma de profundidade correspondentes a uma mesma cena. Ao nosso saber, a integração destes dois tipos de dados na segmentação baseada em curvaturas e a detecção de bordas a partir de mapas de sinais das curvaturas H e K não haviam sido anteriormente realizadas, o que demonstra a originalidade do trabalho realizado. Como resultado destes experimentos, foram desenvolvidos dois novos métodos de detecção de bordas que podem ser aplicados na segmentação de imagens [Silva *et al.*, 2001].

Na outra abordagem, cujos objetivos iniciais eram contornar algumas dificuldades encontradas nos experimentos em detecção de bordas, foram realizados experimentos envolvendo métodos de estimativa robustos para a aproximação de superfícies e extração de regiões de imagens de profundidade [Silva *et al.*, 2002]. Estes trabalhos resultaram no desenvolvimento de um novo método de segmentação por extração de regiões planas utilizando uma versão aperfeiçoada de um método de estimativa robusto e um algoritmo genético [Gotardo *et al.*, 2002]. Os resultados experimentais foram utilizados juntamente com um conjunto de métricas e uma base de imagens comum [Hoover *et al.*, 1996] para avaliar quantitativamente e comparativamente o método desenvolvido.

Os trabalhos aqui apresentados fazem parte de um projeto mais amplo – o Sistema de Recuperação de Imagens por Conteúdo de réplicas digitais em 3D de objetos físicos (SRIC3D) [Vieira *et al.*, 2002] [Vieira, 2002] [Silva, 2002] – em desenvolvimento pelo grupo de pesquisa IMAGO¹. Os resultados obtidos vêm sendo utilizados como suporte a este projeto na extração de características e segmentação de imagens.

1.4 Contribuições

Como resultado dos estudos realizados sobre a utilização de curvaturas de superfícies e métodos de estimativa robustos na segmentação de imagens de profundidade, as seguintes contribuições são apresentadas:

- Indicativo do impacto positivo que a integração de dados de intensidade luminosa exerce na segmentação da imagem de profundidade correspondente à mesma cena.
- Um método inédito de detecção de bordas a partir de mapas de sinais das curvaturas H e K , os quais haviam sido anteriormente utilizados apenas na identificação de regiões, mas não bordas, conforme apresentado neste trabalho.
- Um método de detecção de bordas a partir dos valores da curvatura H calculados para ambas imagens de intensidade luminosa e profundidade, em correspondência.

¹Grupo de Pesquisa em Visão Computacional, Computação Gráfica e Processamento de Imagens da Universidade Federal do Paraná (www.inf.ufpr.br/imagen).

- Um novo método de estimativa robusto agregando informação de orientação de superfície para aplicação em segmentação de imagens de profundidade de modo a evitar alguns casos de incorreto particionamento de superfícies, melhorando as formas das regiões extraídas.
- Um conjunto de parâmetros de algoritmos genéticos para aplicação, juntamente com métodos de estimativa robusta, na extração de superfícies em imagens de profundidade. Os parâmetros apresentados são capazes de acelerar o processo de otimização preservando melhor a diversidade genética da população de cromossomos, o que diminui a possibilidade de convergência precoce.
- Um método de segmentação de imagens de profundidade por extração de superfícies, utilizando o método de estimativa robusto desenvolvido e um algoritmo genético com o novo conjunto de parâmetros estabelecido. Este método de segmentação foi avaliado quantitativamente e comparado positivamente a quatro outros métodos de segmentação muito conhecidos da literatura [Hoover *et al.*, 1996].

1.5 Organização

No restante deste texto, cada uma das abordagens adotadas para a segmentação de imagens de profundidade é relatada em um capítulo distinto, juntamente com os conceitos e trabalhos relacionados, método(s) desenvolvido(s) e discussão dos resultados experimentais.

O capítulo 2, apresenta os conceitos relacionados à segmentação de imagens que são importantes para o entendimento de ambas as abordagens descritas nos capítulos seguintes. Mais especificamente, este capítulo discorre sobre as propriedades dos dois principais tipos de imagens utilizados como dados de entrada, imagens de intensidade luminosa e de profundidade, e sobre seu potencial de integração. Além disso, são apresentadas uma definição detalhada do problema de segmentação de imagens, as principais abordagens empregadas e o método de aproximação de superfícies mais frequentemente utilizado. O capítulo apresenta ainda o método de avaliação utilizado nos experimentos realizados com os métodos desenvolvidos.

No capítulo 3 são descritos os conceitos e trabalhos relacionados aos experimentos em segmentação baseada em curvaturas de superfícies. Primeiramente, são apresentados os principais tipos de curvaturas utilizados e os mais relevantes métodos de segmentação baseados nessas curvaturas. Em seguida são descritos os métodos de detecção de bordas desenvolvidos, cujos resultados experimentais são então apresentados e analisados. O capítulo termina com uma seção de conclusão na qual são ressaltadas as principais considerações sobre os métodos desenvolvidos, bem como os trabalhos futuros.

Os conceitos, trabalhos relacionados e os experimentos realizados em segmentação por extração de superfícies são descritos no capítulo 4. Este capítulo discute inicialmente

alguns dos problemas encontrados pelo método de aproximação de superfícies descrito no capítulo 2 e apresenta os métodos de aproximação robustos e sua aplicação na extração de primitivas geométricas. Em seguida, é apresentada uma análise dos principais métodos de segmentação de imagens de profundidade por extração de superfícies. O restante deste capítulo contém a descrição do método de segmentação desenvolvido, os resultados experimentais e uma análise quantitativa do método baseada nestes resultados. A última seção apresenta a conclusão e os trabalhos futuros para esta segunda abordagem.

As últimas considerações quanto aos trabalhos realizados são encontradas no capítulo 5, que também discute os trabalhos futuros e as perspectivas de integração entre os métodos de detecção de bordas e o de extração de superfícies desenvolvidos.

Capítulo 2

Segmentação de Imagens

Este capítulo tem por objetivo apresentar alguns conceitos fundamentais para a compreensão dos métodos de segmentação de imagens desenvolvidos e dos trabalhos relacionados, apresentados nos demais capítulos.

A seção 2.1 descreve os dois principais tipos de imagens digitais utilizadas em sistemas de visão computacional, enquanto a seção 2.2 discute a integração de diferentes tipos de imagens para melhorar a qualidade do processo de segmentação. O problema da segmentação de imagens é definido e analisado na seção 2.3. A seção 2.4 trata do processo de aproximação de superfícies, realizado durante a segmentação e a seção 2.5 apresenta o método de avaliação utilizado nos experimentos com os métodos desenvolvidos.

2.1 Imagens Digitais

Uma imagem digital é uma representação de uma cena do mundo real e constituída por um conjunto de valores discretizados a partir de um sinal capturado pelo sensor. Esse conjunto de valores é normalmente processado como uma matriz, na qual cada elemento, ou *pixel* (de *picture element*), corresponde a um dos pontos das superfícies em cena. Por este motivo, as imagens digitais também são denominadas *superfícies digitais*. Cada ponto da imagem está associado a um valor (e.g. uma tonalidade de cor ou uma distância) ou a um vetor de valores (e.g. coordenadas (x,y,z) ou componentes de cor RGB). O significado deste valor depende do tipo de sensor que gerou a imagem. A seguir, dois tipos de imagens digitais muito utilizados atualmente, as *imagens de intensidade luminosa* e as *imagens de profundidade*, são apresentados e suas características são discutidas.

2.1.1 Imagens de Intensidade Luminosa

As imagens de intensidade luminosa são aquelas mostradas pelos televisores (em “preto-e-branco” ou a cores). Nestas imagens (figuras 2.1 e 2.2(a)), os dados estão em função da intensidade de luz refletida pelos vários pontos das superfícies em cena. Elas podem ser monocromáticas (e.g. em tons de cinza) ou podem conter informação de cores para cada

ponto. Imagens de intensidade luminosa podem ser utilizadas para distinguir objetos com o mesmo formato, mas com diferentes cores. Ou ainda, para distinguir diferentes posições de um mesmo objeto de formas simétricas (e.g. um cubo com faces de diferentes cores). Estas imagens apresentam um nível de ruído baixo pois a tecnologia empregada em equipamentos como câmeras de vídeo está bastante amadurecida. Além disso, tais equipamentos estão cada vez mais baratos.

Entre os problemas enfrentados na segmentação de imagens de intensidade luminosa, está a dependência em relação à iluminação do ambiente e possíveis variações de cores em uma mesma face de um objeto. Isso faz com que um objeto com diferentes cores, com iluminação irregular ou com sombras sobre suas superfícies pode ser sobre-segmentado. Ou seja, o objeto pode ter uma de suas superfícies incorretamente dividida em duas ou mais regiões distintas.

Além disso, imagens de intensidade luminosa não apresentam informações 3D explícitas pois estas são perdidas na projeção realizada no processo de formação dessas imagens. Somente a localização espacial 2D é dada e um dos maiores problemas é então reconstruir a informação 3D para auxiliar na caracterização dos objetos em cena. A disponibilidade de medidas 3D torna mais fácil (mas não trivial) a determinação de algumas características relacionadas à geometria dos objetos em cena.

Assim, é dificultada a aplicação, nas imagens de intensidade luminosa, de algumas técnicas que requerem informações tridimensionais dos objetos. Muitas dessas técnicas são originalmente desenvolvidas para imagens de profundidade, que vêm sendo cada vez mais utilizadas atualmente e são descritas a seguir.

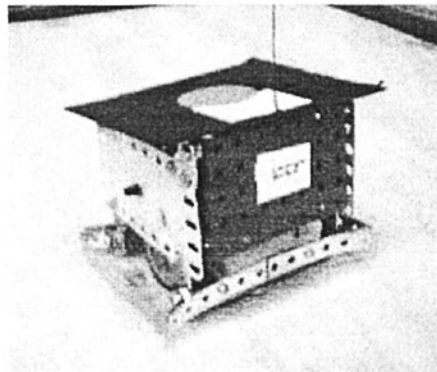
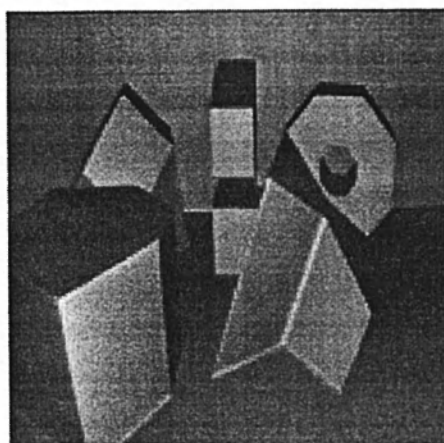


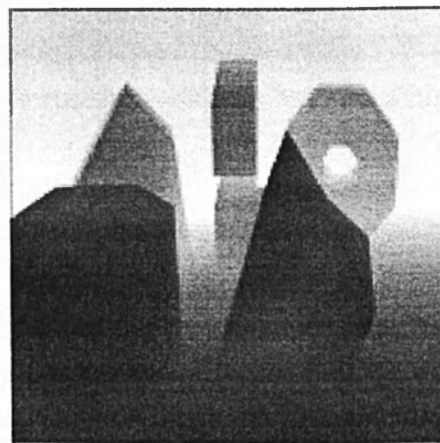
Figura 2.1: Imagem de intensidade luminosa utilizada em projetos de futebol de robôs.

2.1.2 Imagens de Profundidade

Alguns sensores, como rastreadores laser (*laser scanners*), produzem imagens de profundidade, as *range images*. Nestas imagens (figura 2.2(b)), os dados representam as distâncias entre o sensor e os pontos das superfícies na cena observada. Normalmente, após a aquisição, os dados de profundidades são convertidos para um sistema de coordenadas



(a) Imagem de intensidade luminosa



(b) Imagem de profundidade

Figura 2.2: *Dois tipos de imagens para uma mesma cena.*

cartesianas e, assim, a cada ponto da imagem é associado um vetor com as suas coordenadas (x, y, z) .

Deste modo, as imagens de profundidade possuem uma natureza geométrica pois contêm informações 3D explícitas, úteis para a determinação da forma dos objetos. Elas também são denominadas “imagens $2\frac{1}{2}D$ ”, pois contêm coordenadas 3D apenas para os pontos visíveis ao sensor. Ou seja, não é conhecida nenhuma informação sobre as coordenadas dos pontos nas superfícies ocultas dos objetos. O fato de a informação de profundidade ser obviamente relevante para a identificação da forma tridimensional de objetos fez com que muitos trabalhos de pesquisa descartassem totalmente as informações sobre intensidade luminosa. Estas pesquisas têm-se focado basicamente com métodos para descrição de objetos pela reconstrução das superfícies representadas pelos dados de profundidade.

Embora imagens de profundidade venham sendo cada vez mais utilizadas em sistemas de visão, o custo dos sensores que as geram ainda é elevado e estas imagens também são mais sensíveis a ruídos pois a tecnologia empregada na construção dos sensores é relativamente recente e menos desenvolvida que a tecnologia utilizada para obter imagens de intensidade luminosa. Além disso, um sistema que utilize este tipo de imagem não é capaz de distinguir objetos de formas iguais, mas com cores diferentes, nem posições simétricas de um mesmo objeto pois são incapazes de utilizar informações sobre cores e tonalidades destes objetos.

Por exemplo, projetos de futebol de robôs (robótica autônoma multi-agente) empregam sistemas de visão para controlar cada um dos dois times de pequenos robôs que devem competir em um jogo de futebol. Um sensor é posicionado no alto de um “campo” em miniatura e fornece imagens sobre o posicionamento dos robôs, os quais possuem a forma de cubos. O sistema de visão, a partir das imagens de entrada, deve ser capaz de distinguir o time e também determinar a orientação de cada um dos robôs em campo.

Por isso, cada robô possui em sua superfície superior um padrão de cores, uma “camisa”, que identifica o seu time e a sua orientação (figura 2.1). Utilizar imagens de profundidade em tais sistemas seria inadequado pois robôs de diferentes times possuem formas muito semelhantes. Mesmo se fosse possível distingui-los, como estas imagens não dispõem de informações sobre as cores dos objetos, não se poderia identificar a orientação de cada um dos robôs. Assim, os sensores utilizados nestes sistemas são câmeras de vídeo que fornecem imagens de intensidade luminosa coloridas.

No entanto, as informações 3D explícitas apresentadas pelas imagens de profundidade, aliadas ao importante papel das superfícies no processo de reconhecimento de objetos, colaboraram para a popularização do uso desse tipo de imagem em sistemas de visão. Além disso, imagens de profundidades são necessárias em aplicações como, por exemplo, engenharia reversa e inspeção industrial, nas quais características geométricas dos objetos devem ser estimadas de forma precisa [Várady *et al.*, 1997].

2.2 Integração de Imagens

A maioria das pesquisas sobre técnicas de segmentação realizadas atualmente, tem utilizado, como dados de entrada, somente imagens de intensidade luminosa ou de profundidade. Contudo, alguns autores [Besl, 1989] [Besl e Jain, 1985] [Jarvis, 1983] já expressaram suas idéias de que uma melhor segmentação seria obtida caso esses diferentes tipos de dados, para uma mesma cena, fossem combinados.

Nós humanos, temos a habilidade de capturar informações do ambiente através de vários sentidos. Algumas vezes essas informações são complementares e, combinadas, são úteis para eliminar algumas situações envolvendo incógnitas ou ambigüidades.

Câmeras de vídeo fornecem informações sobre a intensidade de luz refletida pelos pontos de uma superfície; rastreadores laser medem as distâncias entre pontos de objetos e o sensor; sensores térmicos captam a temperatura dos objetos em cena; e sensores tácteis fornecem informações sobre os pontos de contato com os objetos. A vantagem de se usar mais de um sensor é clara da observação de que diferentes sensores são sensíveis a diferentes sinais, cada qual revela um conjunto particular de propriedades do ambiente observado.

Atualmente, uma associação que permite uma grande flexibilidade de aplicações é a de informações de intensidade luminosa e de profundidades de uma cena. A escolha deste tipo de associação deve-se em parte ao grande número de trabalhos, encontrados na literatura, que são aplicados a um destes dois tipos de imagens, e, principalmente, a forma extremamente complementar das informações obtidas destes tipos de dados. Deste modo, aumentam-se as chances de eliminar casos de difícil tratamento, normalmente encontrados em aplicações de visão computacional.

Considerando-se a etapa de segmentação, devido a problemas, por exemplo, de iluminação e de resolução, a baixa qualidade das imagens adquiridas determina lacunas de

informações. Este fato é verificado tanto na utilização isolada de imagens de intensidade luminosa quanto de profundidade, dificultando a identificação de elementos característicos na imagem. Como os fatores determinantes destas lacunas afetam os dois tipos de imagens de formas diferentes, a integração dos dados pode aumentar consideravelmente o desempenho da etapa de segmentação. Algumas características podem ser realçadas, gerando redundâncias de informações, o que pode melhorar a precisão dos resultados; outras podem ser compensadas, eliminando as lacunas mencionadas.

Nitzan *et al.* [Nitzan *et al.*, 1977] descrevem um sensor de varredura laser capaz de prover imagens de intensidade luminosa e de profundidade em perfeita correspondência. Isto é, a cada pixel da imagem de intensidade luminosa corresponderá uma profundidade na outra imagem. Este sensor realiza um rastreamento de uma cena com o uso de um feixe laser modulado e, medindo a mudança de fase entre a luz transmitida e a recebida, obtém diretamente os valores de profundidade. Os dados de intensidade luminosa são obtidos medindo a amplitude do raio de luz refletido. Devido ao fato de o feixe laser ser coaxial com a linha de visão, os dados de intensidade luminosa são caracterizados pela inexistência de efeitos de iluminação como sombras e outros reflexos. Assim, tais imagens são denominadas “pseudo-imagens de intensidade luminosa” ou, mais comumente, *imagens de reflectância*. A figura 2.2 contém um exemplo de um par de imagens obtido com um sensor semelhante a este.

2.3 Segmentação de Superfícies Digitais

2.3.1 Definição do Problema

A segmentação de uma imagem pode ser definida como um processo de atribuição de rótulos aos pontos da imagem tal que pontos em uma mesma superfície digital recebam o mesmo rótulo. Ou seja, é um processo de particionamento do conjunto de pontos da imagem digital em subconjuntos, um para cada superfície da cena ou de acordo com outro predicado de homogeneidade.

Seja \mathbf{R} a região correspondente a toda a imagem digital. O objetivo da segmentação é definir n sub-regiões, $\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_n$, tal que

1. $\bigcup_{i=1}^n \mathbf{R}_i = \mathbf{R}$,
2. $\forall i, \mathbf{R}_i$ é uma região conexa,
3. $\forall i \neq j, \mathbf{R}_i \cap \mathbf{R}_j = \emptyset$,
4. $\forall i, P(\mathbf{R}_i) = VERDADEIRO$, e
5. $\forall i \neq j$ e \mathbf{R}_i adjacente a \mathbf{R}_j , $P(\mathbf{R}_i \cup \mathbf{R}_j) = FALSO$,

onde $P(\mathbf{R}_i)$ é um predicado lógico sobre os pontos do conjunto \mathbf{R}_i . Assim, diferentes algoritmos de segmentação podem ser considerados como implementações de diferentes

predicados de homogeneidade. Um predicado de homogeneidade freqüentemente utilizado é aquele cujo valor para um grupo de pontos é verdadeiro se todos estes pontos puderem ser aproximadamente representados por uma única função de superfície.

Um processo de segmentação que identifique com precisão cada região R_i é um requisito fundamental para que um sistema de visão seja capaz de identificar, confiavelmente, as formas e a localização das bordas dos objetos em cena. No entanto, a segmentação de imagens constitui um dos maiores desafios na área de visão computacional pois depende da correta extração de características a partir dos dados de entrada, uma tarefa ainda considerada muito difícil por causa dos ruídos contidos nas imagens do mundo real.

A determinação de quais características dos pontos da imagem são relevantes para a definição do predicado de homogeneidade utilizado também é um fator muito importante e fortemente influenciada pelo domínio da aplicação (conhecimento prévio das cenas a serem analisadas). Uma propriedade desejada para essas características é a invariância a transformações sólidas como rotação e translação. Ou seja, tais características devem-se apresentar equivalentemente nos diversos pontos de vista nos quais um mesmo objeto pode ser visualizado.

Uma abordagem muito utilizada para o reconhecimento de objetos em uma cena consiste em primeiro identificar (segmentar) as suas superfícies distintas, para depois relacioná-las de modo a identificar objetos. Por isso, as características das superfícies dos objetos são utilizadas por grande parte dos métodos de segmentação apresentados na literatura [Besl e Jain, 1988] [Yokoya e Levine, 1989] [Trucco e Fisher, 1992] [Hoover *et al.*, 1996] [Lee *et al.*, 1998] [Silva *et al.*, 2001] [Bellon e Silva, 2002] [Silva *et al.*, 2002]. Essas características são muito importantes porque contêm informações sobre a geometria dos objetos, como, por exemplo, suas formas. Entre as características de superfícies mais utilizadas em métodos de segmentação pode-se citar curvatura e orientação.

Como as imagens do mundo real sofrem degradações denominadas ruídos, identificar corretamente as superfícies de objetos complexo a partir destas imagens ainda é considerada uma tarefa muito difícil para um sistema de visão. Para simplificar um pouco esta tarefa, assume-se que as superfícies contidas nas imagens capturadas são formadas basicamente por poucos modelos de superfícies. Na realidade, as superfícies podem ser formadas por estruturas mais complexas enquanto puderem, em uma pequena vizinhança, ser fielmente aproximadas pelos modelos de superfície utilizados. Para lidar com a complexidade, é preciso então decompor o objeto em partes mais simples e descrever estas partes e a relação entre elas. Se as partes forem complexas, elas podem ser igualmente decompostas do mesmo modo que os objetos maiores. A finalidade do processo de segmentação de imagens é apenas isolar essas partes, que são então representadas pelos modelos de superfície.

Para segmentar as superfícies de uma cena, a abordagem comumente empregada consiste em realizar um procedimento de aproximação de superfícies, no qual são estimados os parâmetros (coeficientes) de um modelo de superfície (e.g. forma implícita

do plano, polinômio biquadrático, splines bicúbicas). Tais parâmetros, os quais descrevem a *instância* do modelo que melhor representa um subconjunto (uma região) dos dados da imagem, são utilizados para calcular características da superfície aproximada.

Primeiramente é preciso determinar quantos e quais modelos serão utilizados na aproximação. Quando vários modelos são utilizados existe a necessidade de decidir qual modelo é o mais apropriado para cada região da imagem, o que está longe de ser considerado trivial. A solução, em muitos casos, é realizar uma aproximação inicial utilizando um modelo simples, como um plano ou polinômio biquadrático, e analisar as características da superfície aproximada (e.g. medida de erro de aproximação) para decidir se é necessário aproximar um modelo de maior ordem e qual o modelo mais apropriado. Porém, em muitos casos é preferível eliminar essa complexidade e empregar um único modelo planar ou biquadrático.

Segundo Wang *et al.* [Wang e Cheng, 1992], a hipótese de modelos de superfícies limitados pode parecer restritiva, mas – mesmo que estruturas de superfícies 3D possam ser arbitrariamente complexas – as superfícies dos objetos encontrados com mais frequência estão restritas a um pequeno domínio. Objetos manufaturados possuem geralmente formas simples e regulares. E como já foi citado, estruturas de superfícies complexas podem ser constituídas por componentes mais simples. Ainda, foi observado [Hakala *et al.*, 1981] que 85% dos objetos industriais podem ser perfeitamente representados ou aproximados por um pequeno número de componentes planos, esféricos, cilíndricos e cônicos.

Mesmo que o modelo utilizado na aproximação seja determinado previamente, para calcular corretamente as características da superfície aproximada é preciso ainda identificar o conjunto de pontos que realmente pertencem a ela. Ou seja, deve-se determinar uma região de pontos conexos que descrevam a mesma superfície, identificando os limites desta região e isolando pontos das demais superfícies dos objetos. Por outro lado, para detectar algumas das bordas de uma superfície [Silva, 2000] [Bellon e Silva, 2002], é necessário aplicar um procedimento de aproximação.

Uma solução para este problema, semelhante ao do ovo e da galinha, é o uso de uma abordagem local, na qual superfícies são aproximadas a pequenas vizinhanças de pontos, possivelmente sobrepostas, e as quais são posteriormente agrupadas, conforme as características das superfícies locais, de modo a formar superfícies maiores. Esta abordagem, empregada na grande maioria dos casos, é denominada *Local-para-Global* e se vale da observação de que na maioria das imagens, os valores de quase todos os pontos estão correlacionados com os valores de seus vizinhos mais próximos. Esta correlação, ou coerência de superfície, nas imagens é originária da coerência espacial das superfícies físicas observadas. Quanto maior a resolução do sensor, um maior número de pontos é definido para cada região da imagem e, portanto, maior é a semelhança entre os valores de um ponto e de seus vizinhos. Filtros de suavização, em geral, também tiram proveito deste fato.

2.3.2 Abordagens

A maioria das técnicas de segmentação encontradas na literatura podem ser classificadas em duas principais categorias: (i) segmentação baseada em superfícies ou regiões; e (ii) segmentação baseada em bordas ou descontinuidades.

Técnicas baseadas em regiões procuram agrupar pontos de características semelhantes para definir regiões conexas da imagem correspondentes a superfícies contínuas. Técnicas de segmentação baseadas na detecção de bordas, por outro lado, procuram identificar pontos que correspondem a descontinuidades nas características das superfícies da imagem, utilizando, assim, contornos para guiar a segmentação da imagem pois delimita, com as bordas detectadas, suas diferentes regiões.

Os algoritmos empregados na segmentação por regiões podem ser divididos em três classes [Gonzalez e Woods, 2000]: crescimento de regiões (*region growing*); divisão-e-união (*split-and-merge*); e agrupamento (*clustering*). Estas classes de algoritmos são brevemente discutidas a seguir, juntamente com a detecção de bordas.

2.3.2.1 Métodos de Agrupamento

Métodos de agrupamento [Celinski e Bellon, 1998] realizam o particionamento dos pontos de uma imagem através da formação de grupos (*clusters*) de pontos com valores semelhantes para um determinado conjunto de características (e.g. tom de cinza e orientação de superfície). Em [Celinski, 1998], Celinski apresenta um estudo comparativo sobre a aplicação de métodos de agrupamento na segmentação de imagens digitais.

Um dos algoritmos de agrupamento mais utilizados é o *K-means*, um método iterativo que estabelece (arbitrariamente ou deterministicamente) um número de centros de regiões e atribui cada ponto da imagem ao centro de maior similaridade. Nas iterações seguintes, novos centros são determinados baseados em informações estatísticas de cada *cluster* e os pontos são então redistribuídos. O processo termina quando não houver alterações nos grupos obtidos entre uma iteração e a seguinte ou quando for alcançado um número máximo de iterações.

2.3.2.2 Crescimento de Regiões

Algoritmos de crescimento de regiões [Besl e Jain, 1988] primeiro determinam pequenas regiões de pontos conexos, chamadas regiões sementes (*seed regions*), com grande confiabilidade quanto ao fato de corretamente representar uma porção de uma superfície. Então, para cada região semente, um modelo de superfície é aproximado. Se o erro de aproximação é grande, o modelo ou a região semente podem ser rejeitados. Caso contrário, por extrapolação, novos pontos na vizinhança da região semente, e que se mostrem compatíveis com a instância do modelo, são anexados e uma nova aproximação é realizada. O processo termina quando não são mais encontrados pontos compatíveis com as regiões semente.

2.3.2.3 Divisão-e-União

Métodos baseados em divisão-e-união [Gonzalez e Woods, 2000] primeiramente participam a imagem em várias e pequenas regiões segundo um critério de homogeneidade aplicado a pequenas vizinhanças de pontos. Em seguida, iterativamente, são unidas regiões adjacentes, as quais sejam estatisticamente prováveis de representar uma mesma superfície. O processo de união termina quando, no conjunto de regiões obtidas, não houver pares de regiões semelhantes.

2.3.2.4 Detecção de Bordas

Bordas são descontinuidades nos valores de uma determinada característica de uma imagem. Os dois tipos de bordas normalmente procurados em imagens digitais são as bordas de profundidade (*step edges* ou *jump edges*) e as bordas de orientação (*roof edges* ou *crease edges*).

Bordas de profundidade indicam descontinuidades significativas entre os valores de profundidade (ou intensidade luminosa) de dois conjuntos de pontos (regiões adjacentes). Bordas deste tipo normalmente delineiam o perfil de objetos e, como mostrado na figura 2.3(a), encontram-se entre superfícies cujo perfil (corte) lembra a forma de um degrau (*step*).

Bordas de orientação, por sua vez, indicam descontinuidades significativas nos ângulos de orientação dos vetores normais às superfícies. Normalmente, estas são as bordas entre superfícies visíveis de um mesmo objeto. O perfil de uma borda de orientação entre duas superfícies, como mostrado na figura 2.3(b), pode lembrar a forma de um telhado (*roof*) de uma casa.

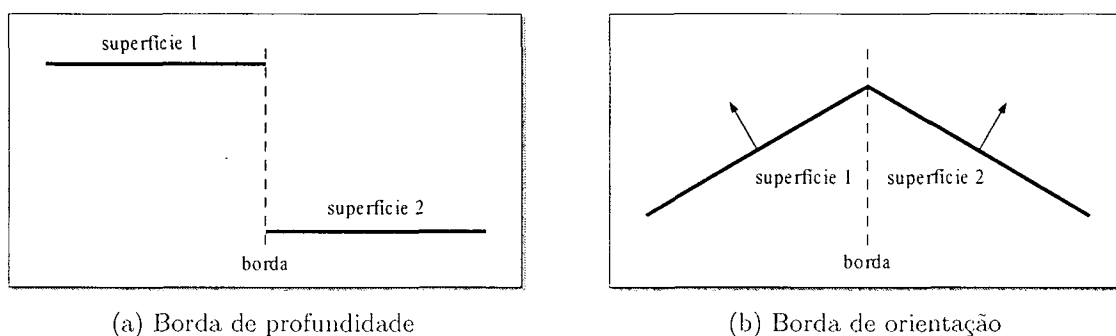


Figura 2.3: *Perfis de bordas.*

Apesar de existirem técnicas bastante eficientes para a detecção de bordas de profundidades, a detecção de bordas de orientação de maneira confiável ainda é um trabalho a ser aperfeiçoado e que vem atraindo o interesse de pesquisadores [Silva, 2000]. Isso porque a determinação correta dos vetores normais às superfícies, em regiões de bordas, é um processo suscetível a erros pois requer a aproximação de uma superfície em uma região contendo pontos de diferentes superfícies.

2.3.2.5 Integrando Bordas e Regiões

Entre os pontos atrativos da abordagem de segmentação por regiões pode-se citar que regiões são elementos grandes da imagem e, por isso, menos sensíveis a degradações causadas por ruídos. Além disso, elas contêm informações sobre a topologia dos objetos, o que pode ser útil para o processo de segmentação. Porém, métodos de segmentação por regiões muitas vezes demandam um tempo computacional grande. Uma dificuldade destes algoritmos está em conseguir isolar corretamente, em regiões de bordas, pontos apresentando características parecidas, mas pertencentes a superfícies distintas e adjacentes.

Métodos de detecção de bordas em geral não garantem a identificação de contornos (mapa de bordas) fechados, necessitando assim de um pós-processamento para realizar o fechamento das bordas e isolar diferentes regiões (superfícies). Em alguns casos, pode ser necessário também realizar um processo de afinamento para obter bordas com apenas um *pixel* de largura. Essa necessidade de refinamento do mapa de bordas inicialmente obtido pode introduzir incorreções no resultado final. No entanto, quando comparada à segmentação por regiões, a segmentação por detecção de bordas mostra-se um processo de menor custo computacional e tende a preservar melhor a localização das bordas e as formas dos objetos em cena.

A princípio, a segmentação completa de uma imagem pode ser obtida utilizando qualquer uma das abordagens acima descritas. Como ao final do processo de segmentação sempre são obtidas ambas, bordas e regiões, estas duas abordagens são consideradas complementares e alguns métodos de segmentação híbridos [Yokoya e Levine, 1989] [Bellon *et al.*, 1999] tiram proveito deste fato integrando técnicas de detecção de bordas a métodos de segmentação por regiões.

2.4 Aproximação de Superfícies

Durante a fase de segmentação, tanto na abordagem por regiões, como na por detecção de bordas, métodos de aproximação de superfícies são aplicados aos dados da imagem digital com o objetivo de definir instâncias de modelos geométricos, de modo a representar da melhor forma possível subconjuntos de pontos da imagem. Essas instâncias, conjuntos de valores para os parâmetros do modelo considerado, são utilizadas então para calcular características de superfície de cada ponto da imagem e identificar as regiões homogêneas correspondentes a cada uma das superfícies da cena.

Além da necessidade de estimar os parâmetros da superfície, outro motivo pelo qual métodos de aproximação são utilizados deve-se ao fato de que os valores fornecidos pelo sensor, para cada ponto da imagem, não estão livres de erros de medição, os quais são denominados de ruídos.

Ruídos podem ser pequenas variações e descontinuidades, introduzidas nos valores

dos pontos da imagem. Eles podem ser resultantes de características físicas do sensor ou do meio de transmissão da imagem, ou ainda devido ao intervalo limitado de valores que podem ser atribuídos a cada ponto da imagem. Esses ruídos, chamados *ruídos de quantização*, fazem com que a imagem de uma estrada lisa e plana aos olhos de uma pessoa, seja mais parecida com uma trilha irregular para uma máquina.

Quando as alterações introduzidas pelos ruídos são de grande magnitude, têm-se os *ruídos impulsivos*, também chamados de ruídos “sal-e-pimenta” por se consistirem de pontos discrepantes, claros ou escuros, localizados em regiões de certa homogeneidade na imagem, conforme mostra a figura 2.4(a). A figura 2.4(b) ilustra (i) o perfil de uma superfície plana ideal, (ii) esse mesmo perfil afetado por ruído de quantização, e (iii) afetado por ruído impulsivo e de quantização.

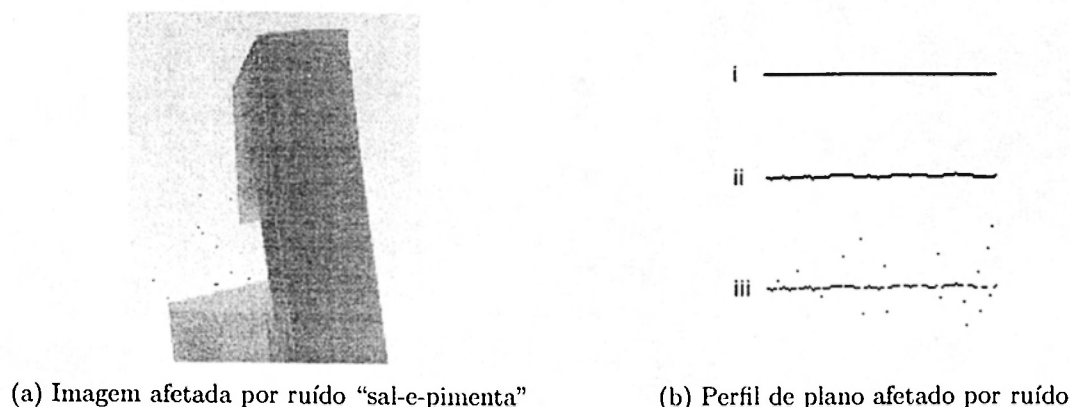


Figura 2.4: *Ruído impulsivo e de quantização.*

Nesse contexto, diferentemente da interpolação de superfícies, a aproximação consiste na identificação de uma *tendência* no conjunto de pontos da superfície, segundo um modelo geométrico. Assim, os parâmetros do modelo são obtidos através de um processo de otimização no qual não é requerido que a superfície aproximada contenha, necessariamente, todos os pontos do conjunto de dados utilizado. Por isso, a aproximação produz um efeito de suavização nos dados de uma superfície, o que, por outro lado, pode causar o borramento de algumas bordas e atrapalhar o processo de segmentação.

Um dos métodos de aproximação mais conhecidos, em várias áreas de pesquisa, é o método dos Mínimos Quadrados [Press *et al.*, 1992], ou *Least Squares* (LS). Para um conjunto de pontos com coordenadas (x_i, y_i) e valor z_i , o LS aproxima uma superfície $\hat{z}(x, y; \hat{\mathbf{a}})$ de modo que os parâmetros estimados, $\hat{\mathbf{a}} = (a_0, a_1, \dots, a_m)$, sejam aqueles que minimizam a função de custo $\chi_{\mathbf{a}}^2$, que é a soma dos quadrados das distâncias ($r_{i,\mathbf{a}} = z_i - \hat{z}(x_i, y_i; \mathbf{a})$) entre cada ponto e a superfície aproximada:

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmin}} \{ \chi_{\mathbf{a}}^2 \} \quad (2.1)$$

onde

$$\chi_{\mathbf{a}}^2 = \sum_i \rho(z_i - \hat{z}(x_i, y_i; \mathbf{a})) \quad (2.2)$$

$$\rho(r_{i,\mathbf{a}}) = r_{i,\mathbf{a}}^2 \quad (2.3)$$

O termo $\rho()$ indica o quanto a distância de cada ponto contribui para o valor da função de custo.

Os parâmetros $\hat{\mathbf{a}}$ que minimizam a função erro podem ser encontrados derivando $\chi_{\mathbf{a}}^2$ em relação a cada a_j , $0 \leq j \leq m$, igualando a zero, e resolvendo [Press *et al.*, 1992] o sistema de equações 2.4.

$$\begin{cases} \frac{\delta \chi_{\mathbf{a}}^2}{\delta a_0} = 0 \\ \frac{\delta \chi_{\mathbf{a}}^2}{\delta a_1} = 0 \\ \vdots \\ \frac{\delta \chi_{\mathbf{a}}^2}{\delta a_m} = 0 \end{cases} \quad (2.4)$$

No entanto, existem vários outros métodos que também podem ser utilizados para resolver aproximações pelo método LS [Lawson e Hanson, 1974]. Alguns dos métodos de segmentação apresentados no próximo capítulo utilizam operadores de convolução, chamados máscaras, para realizar aproximações LS e obter os parâmetros de superfície necessários para o cálculo de curvaturas.

2.5 Comparação de Resultados

Existem diversas características que podem ser consideradas no processo de comparação de resultados da segmentação de imagens. Muitos trabalhos destacam algumas destas características no resultado da segmentação, mas poucos realmente definem métricas para uma comparação destas características comuns entre os segmentadores. Isto é mais evidenciado nos trabalhos que utilizam imagens de profundidade.

Uma dificuldade adicional neste processo está na grande variedade de sensores e tipos de imagens existentes e utilizados pelos grupos de pesquisa em segmentação de imagens de profundidade. A identificação e utilização de um mesmo conjunto de imagens para submeter os algoritmos à avaliação é de grande importância.

No primeiro trabalho específico sobre comparação de segmentadores de imagens de profundidade [Hoover *et al.*, 1996], a avaliação foi feita apenas para imagens com superfícies planares. Como não havia sido apresentado nenhum comparador de segmentadores de imagens de profundidade, os autores limitaram o trabalho a superfícies planares considerando-se que nem esse problema, aparentemente mais simples, tem solução satisfatória. Com o resultado comparativo desse trabalho inicial de comparação, outros trabalhos abrangendo superfícies curvas poderiam ser realizados posteriormente. De fato, alguns grupos de pesquisa já têm concentrado esforços para a criação de métodos de avaliação de algoritmos de segmentação de imagens de profundidade quando aplicados

a imagens com superfícies curvas [Powell *et al.*, 1998].

2.5.1 Método de Avaliação

No trabalho de Hoover *et al.* [Hoover *et al.*, 1996] foram definidos tipos de imagens a serem processadas, um conjunto de métricas e uma ferramenta para avaliação automática dos resultados de segmentadores. Entre as imagens escolhidas para a avaliação estão 40 imagens obtidas com um sensor Perceptron, as quais são utilizadas nos experimentos dos próximos capítulos. Cada imagem contém no máximo 5 (cinco) objetos poliédricos dispostos em diferentes posições. As imagens foram aleatoriamente divididas em dois grupos: 10 imagens de treinamento e 30 imagens de teste. As imagens de treinamento são para o ajuste de limiares dos algoritmos antes de serem submetidos às imagens teste, cujos resultados são avaliados.

Para avaliar os resultados gerados pelos algoritmos, para cada uma das imagens, foram criadas imagens segmentadas manualmente por uma pessoa (*ground truth segmentations*). Cada uma dessas imagens foi revisada por uma segunda pessoa, para eliminar eventuais erros. Finalmente, os reais ângulos entre os vetores normais de cada par de regiões vizinhas foram também disponibilizados.

A avaliação entre a imagem de profundidade segmentada pelo algoritmo (*Machine Segmentation, MS*) e a imagem segmentada manualmente (*Ground Truth, GT*) é feita da seguinte forma:

Seja M o número de regiões em MS , não incluindo pontos não rotulados pelo algoritmo, e N o número de regiões em GT , não incluindo pontos identificados como ruído impulsivo e aqueles para os quais o sensor não conseguiu estabelecer um valor adequado. Seja P_m o número de *pixels* em cada região R_m segmentada pelo algoritmo, onde $m = \{1, \dots, M\}$. Similarmente, seja P_n o número de *pixels* em cada região R_n segmentada manualmente, onde $n = \{1, \dots, N\}$. Seja $O_{mn} = R_m \cap R_n$ o número de *pixels* que possuem as mesmas coordenadas em ambas as regiões R_m e R_n . Então, se não existe sobreposição entre duas regiões, $O_{mn} = 0$, e se existe uma sobreposição total, $O_{mn} = P_m = P_n$.

Uma tabela $M \times N$ é criada, contendo O_{mn} para $m = \{1, \dots, M\}$ e $n = \{1, \dots, N\}$. Implicitamente são anexadas para cada entrada na tabela a percentagem de sobreposição em relação ao tamanho de cada região. $\frac{O_{mn}}{P_m}$ representa a percentagem de m na interseção coberta por m e n . Similarmente, $\frac{O_{mn}}{P_n}$ representa a percentagem de n na interseção coberta por m e n . Estas percentagens são usadas para determinar a classificação das regiões segmentadas. São considerados cinco tipos de classificação para as regiões:

- detecção correta (*correct detection*): usada para as superfícies corretamente segmentadas pelo algoritmo.
- sobre-segmentação (*over-segmentation*): usada no caso de múltiplas detecções de

uma única superfície. o que resulta em uma topologia incorreta.

- sub-segmentação (*under-segmentation*): usada no caso de uma separação insuficiente de múltiplas superfícies que resulta em um subconjunto de topologia correta e geometria deformada.
- perdida (*missed*): usada quando o segmentador não encontra uma superfície que aparece na imagem.
- ruído (*noise*): usada quando o segmentador supõe a existência de uma superfície que não está na imagem.

É importante notar que estas métricas podem ter mais ou menos relevância para diferentes aplicações, por exemplo, na descrição topológica das superfícies, o ruído pode ter menor relevância do que as superfícies perdidas.

As fórmulas para a decisão da classificação são baseadas em uma tolerância dada por um limiar T , onde $0.5 \leq T \leq 1.0$. O valor de T é a percentagem de aceitação de uma determinada classificação, que define a qualidade dos resultados. As métricas definidas são:

1. Deteção correta:

Um par de regiões R_n na imagem GT e R_m na imagem MS são classificadas como deteção correta se:

- $O_{mn} \geq T \times P_m$ (pelo menos $T\%$ dos *pixels* na região R_m da imagem MS são marcados como *pixels* na região R_n da imagem GT), e
- $O_{mn} \geq T \times P_n$.

2. Sobre-segmentação:

A região R_n na imagem GT e um conjunto de regiões na imagem MS , dado por $\{R_{m_1}, R_{m_2}, \dots, R_{m_x}\}$, onde $2 \leq x \leq M$, são classificados como sobre-segmentação se:

- $\forall i \in x, O_{m_i n} \geq T \times P_{m_i}$ (pelo menos $T\%$ dos *pixels* em cada região R_{m_i} na imagem MS são marcados como *pixels* na região R_n na imagem GT), e
- $\sum_{i=1}^x O_{m_i n} \geq T P_n$ (pelo menos $T\%$ dos *pixels* na região R_n da imagem GT são marcadas como *pixels* na união das regiões $\{R_{m_1}, R_{m_2}, \dots, R_{m_x}\}$ na imagem MS).

3. Sub-segmentação:

Um conjunto de regiões na imagem GT dado por $\{R_{n_1}, R_{n_2}, \dots, R_{n_x}\}$, onde $2 \leq x \leq M$ e uma região R_m da imagem MS são classificados como sub-segmentação:

- $\sum_{i=1}^x O_{m_i n} \geq TP_m$ (pelo menos $T\%$ dos *pixels* na região R_m da imagem MS são marcadas como *pixels* na união das regiões $\{R_{n_1}, R_{n_2}, \dots, R_{n_x}\}$ na imagem GT), e
- $\forall i \in x, O_{m_i n} \geq T \times P_{n_i}$ (pelo menos $T\%$ dos *pixels* em cada região R_{n_i} na imagem GT são marcados como *pixels* na região R_m na imagem MS).

4. Perdição:

Uma região R_n na imagem GT que não é participante de nenhuma classificação - detecção correta, sobre-segmentação ou sub-segmentação é classificada como perdição.

5. Ruído:

Uma região R_m na imagem MS que não é participante de nenhuma classificação - detecção correta, sobre-segmentação ou sub-segmentação é classificada como ruído.

Uma vez que todas as regiões estejam classificadas, uma métrica final descreve a precisão da geometria recuperada e é calculada como descrito a seguir.

Cada par de regiões R_{n_1} e R_{n_2} na imagem GT que representam faces adjacentes de um mesmo objeto na cena, tem a diferença angular A_n previamente calculada. Se R_{n_1} e R_{n_2} estão ambos classificados como detecção correta, então o ângulo A_m entre o vetor normal à superfície de suas regiões correspondentes na imagem MS é calculado. O valor absoluto da diferença entre estes dois ângulos é calculado como: $|A_n - A_m|$. Este cálculo é realizado para todas as regiões classificadas como detecção correta. O número de comparações de ângulos, o erro médio e o desvio padrão são armazenados para a segmentação de cada segmentador. Estas medidas fornecem uma estimativa indireta da precisão da geometria recuperada das regiões da imagem corretamente segmentadas.

A ferramenta criada para a comparação automática dos resultados compara a imagem manualmente segmentada e a imagem de regiões gerada pelo segmentador, utilizando as métricas descritas anteriormente. Esta ferramenta foi utilizada para avaliar os resultados de quatro diferentes métodos de segmentação e uma análise comparativa é apresentada em [Hoover *et al.*, 1996].

2.5.2 Algoritmos Comparados

Quatro grupos de pesquisa contribuíram ao projeto de comparação de resultados, avaliando seus próprios algoritmos de segmentação de imagens em regiões planas. Estes algoritmos, identificados pelos nomes das instituições de pesquisa de seus autores, possuem diferenças significativas em suas metodologias e foram avaliados utilizando as métricas acima.

Para possibilitar uma análise comparativa visual, os resultados experimentais dos capítulos 3 e 4 são apresentados juntamente com os resultados obtidos, para as imagens Perceptron, com os algoritmos destes quatro grupos de pesquisa.

Além disso, o método de segmentação por extração de regiões apresentado no capítulo 4 foi avaliado quantitativamente (seção 4.5.4) e comparado às avaliações destes quatro métodos, brevemente descritos a seguir.

2.5.2.1 University of South Florida (USF), EUA

Basicamente, este algoritmo calcula um ajuste planar para cada ponto da imagem e realiza um crescimento de regiões anexando pontos que possuem equações de planos semelhantes. O ponto que apresentar o menor erro residual de aproximação é considerado como ponto inicial para o processo de crescimento de regiões.

Ao final deste processo, todos os pontos estão associados a uma única região. Então, um aproximação de superfície final é realizada para obter a orientação de cada uma das regiões.

2.5.2.2 Washington State University (WSU), EUA

Primeiramente, este algoritmo identifica pontos de bordas de profundidade e, então, calcula os vetores normais às superfícies para cada ponto que não esteja muito próximo a um ponto de borda. Em seguida, pontos com orientação similar são agrupados. Caso os vetores normais de pontos adjacentes, pertencentes a duas regiões distintas, apresentem uma diferença angular média menor que um valor de limiar, essas regiões são unidas.

Por fim, um processo iterativo classifica cada segmento como planar ou não-planar; une segmentos que são adjacentes e possuem parâmetros semelhantes; e realiza uma união dos pontos na fronteira de cada segmento caso o erro de ajuste desses pontos seja inferior a um outro valor de limiar. O processo termina quando não houver mudanças nos rótulos dos pontos.

2.5.2.3 University of Bern (UB), Suíça

O segmentador UB é baseado no fato de que pontos em uma linha de varredura (*scan line*) de uma imagem de profundidade e pertencentes a uma superfície plana formam um segmento linear reto 3D. O algoritmo basicamente divide cada linha de varredura em segmentos lineares e faz um crescimento de regiões agrupando esses segmentos, ao invés de pontos individuais.

O ponto inicial para o crescimento é um conjunto formado por segmentos lineares de três linhas de varreduras vizinhas. Quando não existirem mais segmentos a serem adicionados a esta região, um novo ponto inicial é calculado para começar outro processo de crescimento, formando uma nova região. Este processo se repete até todos os segmentos estarem distribuídos em regiões.

2.5.2.4 University of Edinburgh (UE), Escócia

O primeiro estágio deste algoritmo calcula os vetores normais para cada ponto da imagem utilizando um ajuste de plano. Então, as descontinuidades de profundidade e de orientação são calculadas. No segundo estágio, um mapa de sinais das curvaturas K e H (discutidos no capítulo a seguir) é elaborado.

No estágio seguinte, o algoritmo calcula um ajuste de superfície, baseado no método LS, para cada região da segmentação inicial. Um crescimento de regiões é realizado em três etapas sucessivas – expansão, refinamento e contração – onde são realizados testes para adicionar ou rejeitar um ponto. Estas três etapas se repetem até não haver mais modificações na rotulação dos pontos, ou um número máximo de iterações for alcançado. O quarto e último estágio realiza um refinamento das bordas das regiões empregando um processo semelhante ao utilizado no estágio anterior.

Este método é derivado dos trabalhos de segmentação por curvaturas de superfície de Besl e Jain [Besl e Jain, 1988] e de Trucco e Fisher [Trucco e Fisher, 1992], descritos em maiores detalhes no próximo capítulo.

Capítulo 3

Segmentação Baseada em Curvaturas de Superfícies

O reconhecimento de estruturas tridimensionais a partir de suas projeções bidimensionais é uma área de pesquisa muito importante em visão computacional. Objetos do mundo real são entidades 3D e alguns tipos de imagens são essencialmente 2D, o que implica que informações sobre características das superfícies em cena são perdidas durante o processo de formação dessas imagens. Caracterizar e identificar estruturas 3D a partir de imagens 2D é uma tarefa complexa. Por isso, a maioria dos métodos de segmentação de superfície utiliza como dados de entrada imagens de profundidade contendo informações 3D explícitas e, atualmente, poucos trabalhos de pesquisa tem utilizado características de superfícies obtidas a partir de dados de intensidade luminosa.

As curvaturas das superfícies são utilizadas por vários métodos de segmentação apresentados na literatura [Watson *et al.*, 1985] [Hoffman e Jain, 1987] [Fan *et al.*, 1987] [Besl e Jain, 1988] [Yokoya e Levine, 1989] [Trucco e Fisher, 1992] [Silva *et al.*, 2001] pois são características invariantes a transformações geométricas sólidas (e.g. mudanças no ponto de visão). Essa invariância contribui para que sistemas de visão possam reconhecer objetos dispostos em diferentes configurações. O algoritmo de classificação e segmentação de superfície introduzido em [Besl e Jain, 1988], e utilizado por outros métodos [Yokoya e Levine, 1989] [Silva *et al.*, 2001], é baseado somente no conhecimento de superfícies digitais, podendo ser aplicado tanto a imagens de profundidade quanto a imagens de intensidade luminosa.

Imagens de intensidade luminosa podem ser consideradas superfícies digitais quando seus dados apresentam a propriedade de coerência de superfície, no sentido de que os dados podem ser interpretados como amostras de uma função de superfície. Como a diferença entre imagens de profundidade e de intensidade luminosa é a interpretação dos valores medidos pelo sensor para cada *pixel*, a diferença na aplicação do algoritmo em cada tipo de imagem relaciona-se a como os resultados da segmentação são interpretados. Estes resultados são úteis quando as superfícies da imagem de intensidade

luminosa correspondem às superfícies físicas observadas em uma cena. Essa suposição é normalmente feita em métodos de detecção de bordas, nos quais considera-se que as bordas das superfícies de imagens de intensidade luminosa correspondem às bordas das superfícies físicas na cena.

Neste capítulo, são apresentadas duas abordagens originais para detecção de bordas. As abordagens desenvolvidas integram informações de curvaturas de superfícies obtidas de ambas imagens de profundidade e de intensidade luminosa correspondentes a uma mesma cena. Os mapas de bordas produzidos podem ser utilizados para melhorar a qualidade do processo de segmentação de imagens de profundidade. A seção 3.1 descreve os tipos de curvaturas mais utilizados na segmentação de imagens, além de um método de classificação de superfícies baseado em curvaturas. Os trabalhos relacionados a segmentação de imagens baseada em curvaturas de superfícies são descritos na seção 3.2. As duas novas abordagens desenvolvidas são apresentadas na seção 3.3, juntamente com os resultados experimentais.

3.1 Curvaturas de Superfícies

Em qualquer ponto de uma superfície, pode-se escolher dois vetores ortogonais, no plano tangente, e analisar-se o comportamento da superfície nestas duas direções. Cada direção determina uma curva na superfície. A cada curva está associado um valor, denominado *curvatura normal*, medindo a tendência com a qual a curva se afasta do plano tangente. Escolhendo-se apropriadamente entre todos os pares de vetores ortogonais do plano tangente a um determinado ponto p da superfície, encontra-se dois valores de curvatura para p , um dos quais é o maior e o outro é o menor possível. Estes valores, respectivamente k_1 e k_2 , são chamados de *curvaturas principais* e estão associadas às duas *direções principais*.

Estimando-se esses dois valores de curvatura para um determinado ponto na imagem e analisando-se apenas o sinal (negativo, positivo ou zero) dos valores estimados, pode-se classificar este ponto como pertencente a um dentre seis tipos de superfícies: planar (*flat*), esférica convexa (*peak*), esférica côncava (*pit*), cilíndrica convexa (*ridge*), cilíndrica côncava (*valley*) e superfície em forma de sela (*saddle*). Essa classificação é feita como mostra a tabela 3.1.

$k_2 \backslash k_1$	$k_1 < 0$	$k_1 = 0$	$k_1 > 0$
$k_2 < 0$	peak	ridge	saddle
$k_2 = 0$	ridge	flat	valley
$k_2 > 0$	saddle	valley	pit

Tabela 3.1: Classificação de superfícies utilizando as curvaturas principais.

Outros dois tipos de curvaturas utilizados com maior frequência na classificação de superfícies são as curvaturas *Média* (curvatura H) e *Gaussiana* (curvatura K), que podem

ser definidas em função das curvaturas principais:

$$H = \frac{k_1 + k_2}{2} \quad K = k_1 \times k_2 \quad (3.1)$$

A vantagem em relação ao uso das curvaturas principais é que as curvaturas H e K possibilitam a distinção de um maior número de tipos de superfícies e podem ser calculadas sem necessitar a identificação das direções principais. Os tipos de superfícies identificados pelos sinais das curvaturas H e K são mostrados na figura 3.1 e a classificação é feita de acordo com a tabela 3.2.

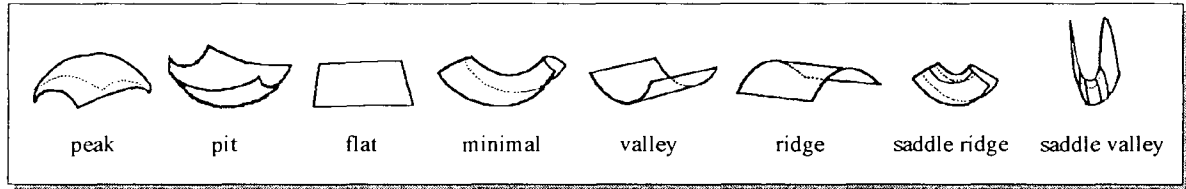


Figura 3.1: Superfícies classificadas pelos sinais das curvaturas H e K .

$H \backslash K$	$K > 0$	$K = 0$	$K < 0$
$H < 0$	peak	ridge	saddle ridge
$H = 0$	–	flat	minimal
$H > 0$	pit	valley	saddle valley

Tabela 3.2: Classificação de superfícies utilizando as curvaturas H e K .

Calculando-se os valores dessas duas curvaturas para cada ponto da imagem, pode-se classificar esse pontos de acordo com a tabela acima e atribuir-lhes um rótulo (valor) que identifica o tipo de superfície em que estão localizados. O conjunto de pontos rotulados forma uma imagem de rótulos denominada *mapa de sinais HK* (*HK-sign map*) [Besl e Jain, 1988]. Este mapa de sinais é considerado como um resultado intermediário da segmentação (uma segmentação rústica) e nele é possível identificar algumas regiões de pontos pertencentes a um mesmo tipo de superfície.

Nos cálculos utilizados para a obtenção dos valores de curvatura, dificilmente os valores resultantes são precisamente zero. Assim, uma dificuldade encontrada no processo de classificação segundo os sinais das curvaturas é estabelecer valores de limiar adequados para determinar o quão próximo de zero os valores de curvatura serão considerados como tal. Como a curvatura H é dada pela média aritmética das curvaturas k_1 e k_2 , seu valor é menos sensível a ruídos do que a curvatura K , determinada pelo produto das curvaturas principais. Logo, faz-se necessário o uso de dois valores de limiar distintos, um para a curvatura H (ε_H) e outro para a curvatura K (ε_K). Tais limiares variam de acordo com o tipo de imagem de entrada e, por isso, têm sido geralmente estabelecidos empiricamente, embora alguns trabalhos [Cai, 1989] [Hilton *et al.*, 1995] tenham sido realizados na tentativa de determinar esses valores automaticamente a partir dos dados da imagem.

A análise dos sinais é feita como mostra a equação 3.2. Para calcular os rótulos de superfície (os valores do mapa de sinais) utiliza-se a fórmula da equação 3.3. Pela definição, os valores atribuídos como rótulos vão de 1 a 9, excluindo 4, e correspondem à leitura da tabela 3.2 da esquerda para a direita e de cima para baixo.

$$\text{senal}_\epsilon(x) = \begin{cases} 1 & \text{se } x > \epsilon_{h,k} \\ 0 & \text{se } |x| \leq \epsilon_{h,k} \\ -1 & \text{se } x < -\epsilon_{h,k} \end{cases} \quad (3.2)$$

$$\text{rótulo}_{HK} = 1 + 3(1 + \text{senal}_\epsilon(H)) + (1 - \text{senal}_\epsilon(K)) \quad (3.3)$$

3.1.1 Cálculo das Curvaturas

Seja f uma função de superfície e $f(x, y)$ o valor de intensidade luminosa ou de profundidade para o *pixel* de coordenadas (x, y) . Se para uma vizinhança de pontos $(x, y, f(x, y))$ forem estimados os parâmetros de f , através de uma aproximação de superfície local, os valores das curvaturas H e K no ponto central dessa região podem ser definidos como:

$$H = \frac{f_{xx} + f_{yy} + f_{xx}f_y^2 + f_{yy}f_x^2 - 2f_xf_yf_{xy}}{2(1 + f_x^2 + f_y^2)^{\frac{3}{2}}} \quad K = \frac{f_{xx}f_{yy} - f_{xy}^2}{(1 + f_x^2 + f_y^2)^2} \quad (3.4)$$

onde f_x, f_y, f_{xx}, f_{yy} e f_{xy} são as derivadas parciais de f .

A abordagem mais utilizada para o cálculo das curvaturas H e K é constituída por métodos analíticos [Flynn e Jain, 1989]. Estes métodos (1) realizam uma aproximação de superfície local, na vizinhança do ponto de interesse; (2) calculam, a partir dos parâmetros de superfície estimados, os parâmetros das funções de derivada parcial; e (3) avaliam os valores das curvaturas H e K , no ponto, utilizando os valores das derivadas parciais e as fórmulas da equação 3.4. A diferença entre os diversos métodos analíticos consiste no modelo de superfície e no método de aproximação utilizados por cada um.

O método [Besl e Jain, 1988] freqüentemente utilizado para o cálculo das curvaturas H e K realiza aproximações Least Squares (LS) locais de uma superfície polinomial biquadrática:

$$f(x, y) = a_0 + a_1x + a_2y + a_3xy + a_4x^2 + a_5y^2 \quad (3.5)$$

Para cada ponto da imagem, os cinco valores das derivadas parciais da função de superfície, naquele ponto, são obtidos com operações de *convolução* de máscaras (matrizes) binomiais quadradas. Nestas operações, uma máscara é “sobreposta” a uma região da imagem e cada elemento seu é multiplicado pelo valor do ponto subjacente. A soma dessas multiplicações é o resultado da operação para o ponto localizado no centro da máscara. Portanto, para cada ponto da imagem em que a máscara for centrada, a operação de convolução também utilizará os dados de outros pontos na vizinhança.

As máscaras binomiais (equações 3.6) utilizadas no cálculo das derivadas parciais são resultantes da multiplicação de vetores obtidos de polinômios ortogonais (equações 3.7).

$$\begin{aligned} D_x &= d_0 d_1^T & D_y &= d_1 d_0^T \\ D_{xx} &= d_0 d_2^T & D_{yy} &= d_2 d_0^T & D_{xy} &= d_1 d_1^T \end{aligned} \quad (3.6)$$

$$\begin{aligned} d_0 &= \frac{1}{7} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T \\ d_1 &= \frac{1}{28} \begin{bmatrix} -3 & -2 & -1 & 0 & 1 & 2 & 3 \end{bmatrix}^T \\ d_2 &= \frac{1}{84} \begin{bmatrix} 5 & 0 & -3 & -4 & -3 & 0 & 5 \end{bmatrix}^T \end{aligned} \quad (3.7)$$

Deste modo, a convolução da máscara D_x em uma imagem resultará em outra imagem contendo os valores de f_x para cada um dos pontos dessa imagem de entrada. Analogamente, as convoluções de D_y , D_{xx} , D_{yy} e D_{xy} resultarão nas imagens para f_y , f_{xx} , f_{yy} e f_{xy} , respectivamente, como mostrado nas seguintes equações:

$$\begin{aligned} f_x &= D_x * f & f_y &= D_y * f \\ f_{xx} &= D_{xx} * f & f_{yy} &= D_{yy} * f & f_{xy} &= D_{xy} * f \end{aligned} \quad (3.8)$$

onde o símbolo “*” representa a operação de convolução.

Uma vez calculadas as curvaturas H e K , ou as curvaturas principais (i.e. utilizando outro método), o outro par de curvaturas pode ser obtido pela relação

$$k^2 - 2Hk + K = 0 \quad (3.9)$$

onde k_1 e k_2 são as raízes da equação.

Utilizando os valores das derivadas parciais, já estimados, pode-se calcular também valores de outras características das superfícies da imagem, a um custo computacional muito baixo. Por exemplo, pode-se calcular diretamente as seguintes características:

- Coeficientes do vetor normal à superfície em cada ponto: $n = \frac{1}{\sqrt{1 + f_u^2 + f_v^2}} \begin{bmatrix} -f_u \\ -f_v \\ 1 \end{bmatrix}$
- Ângulo de orientação do vetor normal em cada ponto: $\alpha = \tan^{-1} \left(\frac{f_u}{f_v} \right)$
- Ângulo de inclinação do vetor normal em cada ponto: $\Theta = \tan^{-1} \left(\sqrt{f_u^2 + f_v^2} \right)$

Uma observação em relação a métodos analíticos para o cálculo de curvaturas de superfícies – e de outras características, como apresentado nesta seção – é que os valores calculados para elas são intensamente sensíveis a ruídos pois faz-se necessário estimar derivadas da função de superfície, nas quais os efeitos dos ruídos são amplificados [Besl e Jain, 1988]. Por isso, a literatura apresenta um grande número de trabalhos

[Flynn e Jain, 1989] que buscam desenvolver métodos de cálculo de curvatura mais confiáveis.

3.2 Trabalhos Relacionados

No reconhecimento de objetos a partir de uma imagem, dois tópicos são muito importantes: a caracterização de superfícies e a segmentação de imagens. Nos últimos anos, vários trabalhos relacionados a estes assuntos vêm sendo desenvolvidos.

Na seção 2.3.2 foi visto que existem duas abordagens distintas (e complementares) que podem ser adotadas para o processo de segmentação. Uma dessas abordagens procura por semelhanças nas características dos pontos da imagem visando à formação de regiões homogêneas. A outra procura por linhas de descontinuidades nessas características para isolar os pontos de diferentes regiões.

A seguir, será realizada uma revisão de alguns trabalhos relacionados à utilização de curvaturas de superfície no processo de segmentação baseada em regiões ou bordas.

3.2.1 Segmentação Baseada em Regiões

3.2.1.1 Besl e Jain

Besl e Jain descrevem [Besl e Jain, 1988] o primeiro método de segmentação, baseado em regiões, que faz uso de mapas de sinais das curvaturas Média e Gaussiana. O algoritmo é composto por três estágios e pode ser aplicado a qualquer imagem que possua a propriedade de coerência de superfície, conforme descrita no início deste capítulo.

O primeiro estágio do algoritmo realiza os seguintes procedimentos: (1) estima a variância do ruído presente na imagem para verificar se a condição de coerência de superfície é satisfeita; (2) realiza uma suavização na imagem utilizando um filtro Gaussiano [Jain *et al.*, 1995]; (3) estima as derivadas parciais para as superfícies suavizadas através de uma aproximação de superfície biquadrática (utilizando o método LS) a uma pequena vizinhança de pontos; e (4) calcula os valores das curvaturas Média e Gaussiana e o mapa de sinais HK.

O segundo estágio, iterativo, (1) identifica as regiões conexas formadas por pontos com o mesmo rótulo de superfície; (2) encontra pequenas regiões interiores, chamadas sementes (*seeds*), nas quais a classificação dos pontos é mais confiável; e (3) para cada região semente, realiza um procedimento iterativo de crescimento de regiões baseado na aproximação de superfícies utilizando alguns modelos diferentes (*variable-order surface fitting*), de acordo com o rótulo da região.

O terceiro estágio, não iterativo, (1) combina os descritores de regiões para obter um descritor de região unificado para cada superfície, (2) organiza um grafo de adjacência de superfícies, e (3) une primitivas de superfície adjacentes e compatíveis, formando regiões maiores e produzindo a segmentação final da imagem.

Dentre as vantagens deste método de segmentação, ressalta-se que ele não se limita ao reconhecimento de superfícies planas e pode ser aplicado a qualquer tipo de imagem que apresente a propriedade de coerência de superfície. Foram obtidos bons resultados na utilização deste método para a segmentação, isolada, tanto de imagens de profundidade, quanto de intensidade luminosa [Besl, 1988]. Porém nenhuma integração entre esses dois tipos de imagens foi realizada. Como desvantagens, ressalta-se a utilização do método LS, muito sensível a ruídos, no cálculo das derivadas parciais e os efeitos colaterais causados nas bordas dos objetos pelo filtro Gaussiano. Estes dois pontos negativos acabam afetando a qualidade do mapa de sinais HK e, por sua vez, das regiões sementes.

Como veremos adiante, muitos métodos de segmentação desenvolvidos posteriormente foram fortemente influenciados por este método.

3.2.1.2 Yokoya e Levine

Yokoya e Levine propõem [Yokoya e Levine, 1989] uma abordagem híbrida para o problema de segmentação de imagens de profundidade, combinando técnicas de segmentação por regiões com técnicas de detecção de bordas. A imagem de profundidade de uma cena é dividida em primitivas de superfícies não contendo descontinuidades de profundidade e de orientação. No entanto, as bordas não são formadas a partir de informações de curvaturas. Somente as regiões o são.

Este método realiza uma aproximação LS adaptativa, de um polinômio biquadrático, nas proximidades de cada ponto da imagem para obter as derivadas parciais e calcular as curvaturas H e K. Essa abordagem adaptativa tem a finalidade de obter melhores estimativas de curvaturas nas regiões próximas às bordas dos objetos, onde geralmente os erros de aproximação são grandes devido a presença de pontos de diferentes superfícies (*pseudo-outliers*). A abordagem consiste em realizar várias aproximações, utilizando diferentes vizinhanças quadradas, de pequenas dimensões e centradas próximas ao ponto de interesse. Os parâmetros finais são aqueles resultantes da aproximação que gerou menor valor de erro, igual ao somatório dos resíduos de todos os pontos da vizinhança utilizada. Então, são calculados as curvaturas H e K e o mapa de sinais.

Também são calculadas duas outras segmentações iniciais, baseadas em bordas de objetos. Uma contém *bordas de orientação*, as bordas entre duas ou mais superfícies visíveis de um mesmo objeto, que são identificadas a partir de descontinuidades na orientação dos vetores normais às superfícies, os quais são obtidos das derivadas parciais da superfície. A outra contém *bordas de profundidade*, as bordas formando o perfil dos objetos em cena, que são detectadas através da análise de descontinuidades de grande magnitude nos valores de profundidade da imagem. As três segmentações iniciais são então combinadas para produzir a segmentação final da imagem de profundidade.

A união de mapas de bordas ao mapa de sinais HK é um fator positivo, apresentado por este método, para a segmentação das superfícies da imagem. Muitas vezes, superfícies

vizinhas de um objeto podem receber o mesmo rótulo e acabar formando uma única região no mapa de sinais. Com as informações de bordas agregadas por este método, estas superfícies são mais facilmente segmentadas.

A abordagem adaptativa utilizando o LS fornece melhores resultados que aqueles obtidos pela aproximação LS convencional. Mesmo assim, devido à sensibilidade do LS, as aproximações ainda são corrompidas nos casos em que todas as vizinhanças utilizadas contêm *outliers*.

3.2.1.3 Trucco e Fisher

Em [Trucco e Fisher, 1992], Trucco e Fisher apresentam um sistema que produz uma descrição qualitativa baseada em superfícies para uma cena de uma imagem de profundidade. Este método, assim como o de Besl e Jain, também se baseia em sinais das curvaturas Média e Gaussiana para a classificação dos pontos nas superfícies.

Inicialmente, é realizada uma detecção de bordas para que, posteriormente, os efeitos colaterais da suavização da imagem – como a alteração dos sinais das curvaturas, próximo às bordas – sejam atenuados. A suavização é então realizada adaptativamente através de condições que preservam a forma dos objetos. Na seqüência, para cada ponto, são estimados os sinais para as curvaturas Média e Gaussiana a partir da aproximação local de uma *spline* [Jain *et al.*, 1995] às superfícies. Então, antes de calcular o mapa de sinais HK, são montadas imagens identificando os sinais das curvaturas em cada ponto. A estas imagens são aplicadas operações de morfologia matemática [Sonka *et al.*, 1998] (erosão e dilatação) para a eliminação de pequenas regiões, formadas principalmente por ruídos. Após este processo, cada ponto é rotulado conforme os sinais resultantes.

Com os melhoramentos introduzidos neste método, o mapa de sinais obtido – que no método de Besl era apenas uma segmentação grosseira e intermediária – consiste na segmentação final. Entretanto, experimentos realizados constataram que os valores das curvaturas são subestimados devido ao processo de suavização, concluindo que é mais fácil estimar, com precisão, os sinais de curvaturas do que seus valores.

3.2.2 Segmentação por Detecção de Bordas (Fan *et al.*)

Na abordagem de Fan *et al.* [Fan *et al.*, 1987], as bordas, características explícitas das superfícies de objetos tridimensionais, são detectadas para produzir a segmentação das superfícies de imagens de profundidade. As bordas dos objetos são localizadas através da identificação dos pontos nos quais os valores de uma das curvaturas principais mudam de sinal (*zero-crossings*) ou atingem valores extremos (máximos e mínimos locais), picos.

A figura 3.2 mostra cortes transversais de regiões de borda entre superfícies e o comportamento dos valores das curvaturas nessas regiões. Em bordas de profundidade, existe uma mudança de sinal e dois picos de sinais opostos (figura 3.2(a)). Em bordas de orientação, os valores das curvaturas exibem um pico (figura 3.2(b)). Em bordas formadas

por uma superfície côncava e um plano, os valores das curvaturas também exibem um pico (figura 3.2(c)). Para bordas entre uma superfície côncava e uma convexa, a resposta dos valores das curvaturas é um pico e uma mudança de sinal (figura 3.2(d)).

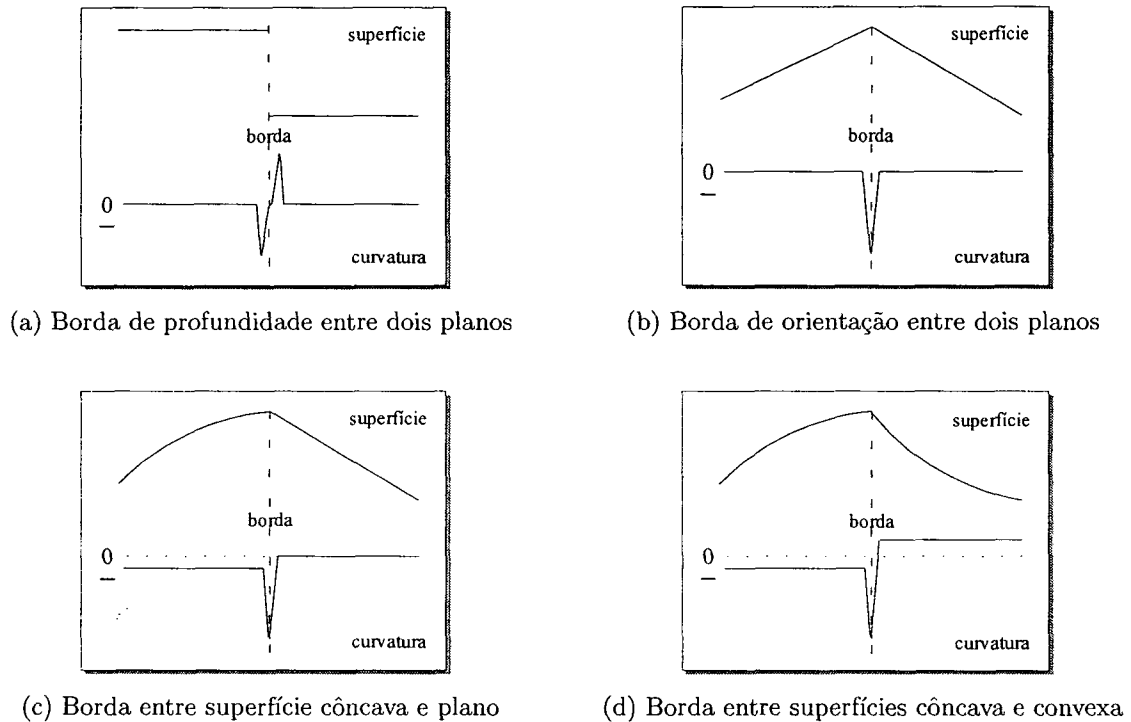


Figura 3.2: Comportamento dos valores de curvatura.

Após o processo de detecção de bordas, as regiões resultantes são então aproximadas por um modelo de superfície biquadrático utilizando o método LS.

Este método de segmentação foi o primeiro a utilizar medidas de curvatura para identificar bordas, ao invés de regiões. Contudo, identificar máximos locais é uma tarefa difícil pois depende da correta estimação dos valores de curvatura, os quais, como apontado por Trucco *et al.* geralmente são subestimados, devido a suavizações realizadas nos dados da imagem, e são mais difíceis de serem obtidos com precisão do que os seus sinais.

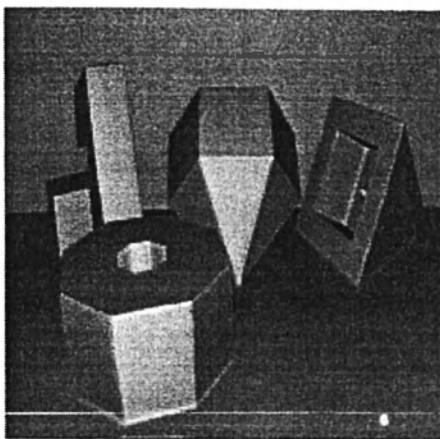
3.3 Métodos Desenvolvidos

Devido à importância do processo de segmentação no reconhecimento de objetos por sistemas de visão computacional, e devido ao fato de ainda não existir uma solução definitiva para o problema, o objetivo principal deste trabalho é apresentar contribuições à área de pesquisa através do desenvolvimento de melhorias a métodos existentes para a segmentação imagens de profundidade. As características desejadas para os métodos resultantes são: robustez à presença de ruídos, normalmente encontrados nas imagens do mundo real; e preservação da topologia dos objetos, localizando de forma precisa as bordas entre as diferentes superfícies em cena.

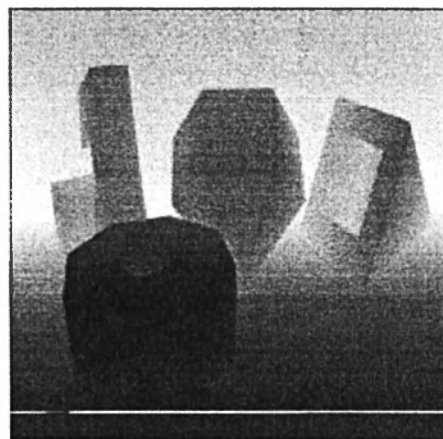
Uma abordagem escolhida para desenvolver métodos de segmentação mais robustos foi a integração, inédita, de informações das curvaturas de superfícies H e K calculadas a partir de imagens de profundidade e de intensidade luminosa correspondentes à mesma cena. A motivação para isso resulta do fato de que resultados adequados a algumas aplicações foram obtidos por métodos de segmentação baseados nas curvaturas de superfícies H e K quando aplicados a ambos os tipos de imagens, porém isoladamente. Ou seja, nenhuma integração dos resultados obtidos para os dois tipos de dados de entrada foi realizada.

Possuir um maior conjunto de informações, complementares, sobre a cena em análise pode proporcionar ao método muitas vantagens, como torná-lo mais flexível e confiável. O algoritmo pode ser até mais simples e a execução do processo mais rápida. Mesmo que o maior conjunto de dados de entrada implique um maior custo computacional, conforme as arquiteturas de computadores e sensores evoluem, essa integração acaba por se tornar uma abordagem atraente para o desenvolvimento de sistemas de visão computacional.

Nesta seção, são apresentados dois novos métodos de detecção de bordas que foram desenvolvidos e que podem ser aplicados na segmentação de imagens de profundidade. Estes métodos são baseados em informações sobre as curvaturas H e K , obtidas a partir de um par de imagens de entrada, uma de profundidade e outra de intensidade luminosa, correspondentes a uma mesma cena. A figura 3.3 mostra um exemplo de par de imagens de entrada, obtidas com um sensor *Perceptron*, que podem ser utilizadas. Um método original de detecção de bordas a partir de mapas de sinais HK [Silva *et al.*, 2001] é descrito na seção 3.3.1. A seção 3.3.2 contém a descrição do outro método de detecção de bordas proposto, o qual é baseado apenas nos sinais dos valores estimados para a curvatura H .



(a) Imagem de intensidade luminosa



(b) Imagem de profundidade

Figura 3.3: Exemplo de imagens obtidas com um sensor *Perceptron* (*perc.test.26*)

3.3.1 Detecção de Bordas em Mapas de Sinais HK

O método de detecção de bordas apresentado nesta seção produz mapas de bordas a partir de mapas de sinais HK calculados para um par de imagens de entrada, uma de intensidade luminosa e outra de profundidade, correspondentes à mesma cena. Como visto na seção 3.2, mapas de sinais HK têm sido utilizados somente para a identificação e classificação de regiões de pontos em um mesmo tipo de superfície. No entanto, esses mapas de sinais não haviam sido anteriormente utilizados para identificar bordas entre as superfícies de objetos, o que mostra a originalidade da abordagem descrita nesta seção.

O método desenvolvido recebe como dados de entrada um par de imagens como o da figura 3.3. Conforme mostra a figura 3.4, inicialmente, uma mesma seqüência de estágios de processamento é aplicada independentemente a cada imagem, produzindo dois mapas de bordas intermediários. Um último estágio consiste na integração que produz o mapa de bordas final. Cada um destes estágios de processamento é visto em detalhes a seguir.

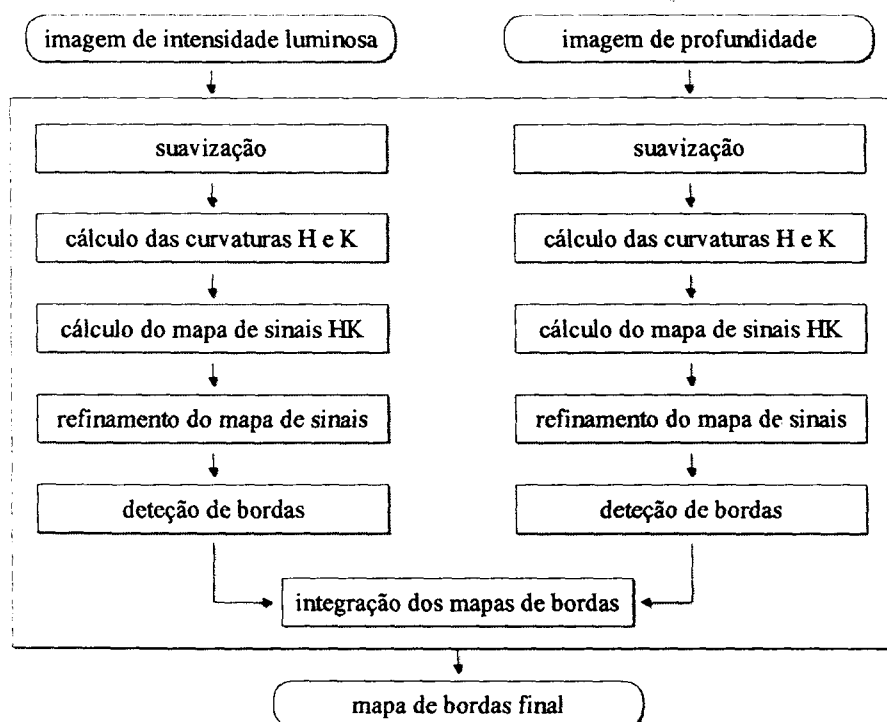


Figura 3.4: Diagrama de estágios de processamento do método de detecção de bordas.

3.3.1.1 Suavização

O objetivo do processo de suavização inicialmente aplicado a cada imagem de entrada é atenuar os efeitos indesejados introduzidos pelos ruídos da imagem no cálculo das curvaturas H e K, o estágio seguinte, muito sensível a ruídos.

Ruídos de quantificação, os quais ocorrem devido a imprecisões do sensor e também por causa do número limitado de valores que podem ser atribuídos por este a cada ponto da imagem, afetam a continuidade dos valores de pontos de uma mesma superfície da

imagem e, desta forma, interferem no processo de estimação dos valores de curvatura em cada ponto. A suavização por um filtro *Gaussiano* [Jain *et al.*, 1995] é uma técnica normalmente utilizada para atenuar esse tipo de ruído e pode ser implementada por uma máscara de convolução cujos coeficientes (pesos), definidos de acordo com a forma de uma função Gaussiana, são menores para os pontos vizinhos mais distantes do ponto central e maiores para aqueles mais próximos. A máscara de suavização, S , utilizada em [Besl e Jain, 1988], e também no método desta seção, é definida como $S = \bar{s} \bar{s}^T$, onde o vetor coluna \bar{s} é mostrado abaixo:

$$\bar{s} = \frac{1}{64} [1 \ 6 \ 15 \ 20 \ 15 \ 6 \ 1]^T \quad (3.10)$$

Assim, a suavização é realizada pela convolução desta máscara S , de dimensões 7×7 , centrada em cada ponto da imagem. Um dos efeitos colaterais de filtros de suavização como este, do tipo por difusão, é a suavização de bordas bem definidas, o que pode ser prejudicial para a sua detecção e localização. Este fato é ilustrado na figura 3.5, que mostra o perfil de uma região de borda, entre dois planos, antes e depois da suavização.

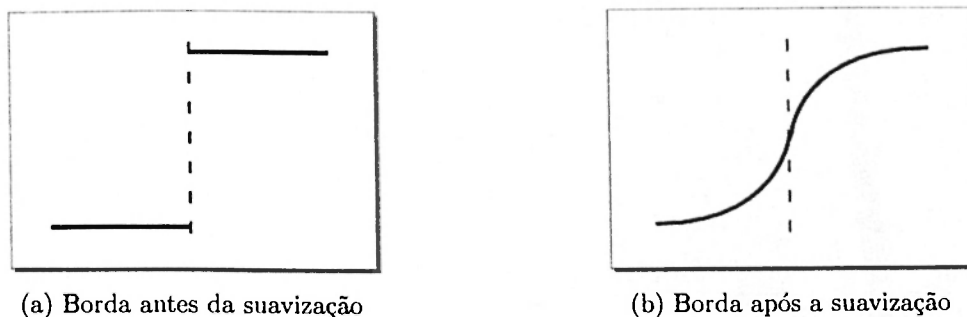


Figura 3.5: Efeito de um filtro de suavização *Gaussiano* nas bordas de uma imagem.



Figura 3.6: Aplicação de filtro mediano para eliminar ruído impulsivo nas imagens.

No entanto, imagens do mundo real não são afetadas apenas por ruídos de quantização. Elas também podem apresentar degradações correspondentes a ruídos impulsivos, os

ruídos do tipo “sal-e-pimenta” (figura 3.6(a)). Filtros Gaussianos não são capazes de eliminar tal tipo de ruído e pontos ruidosos acabam sendo “borrados”, afetando pontos vizinhos e criando pequenas regiões de ruído (figura 3.6(b)). De modo a diminuir o nível de ruído impulsivo na imagem, antes de realizar a suavização com o filtro Gaussiano, aplica-se primeiro uma filtragem por *mediana* (*Median*) com uma máscara de dimensões 3×3 . Estes filtros são capazes de atenuar o nível de ruído impulsivo, preservando as linhas das imagens (e.g. bordas). Num filtro por mediana, o valor do ponto central da máscara utilizada é substituído pela mediana do conjunto de valores dos pontos vizinhos na área da máscara. Com a aplicação do filtro por mediana, antes do Gaussiano, foram obtidos melhores resultados quanto à redução de ruídos (figura 3.6(c)).

3.3.1.2 Cálculo das curvaturas H e K

Este estágio de processamento recebe como entrada a imagem resultante do processo de suavização e produz duas imagens de saída, uma com valores da curvatura H e outra com valores da curvatura K para cada ponto da imagem. O método escolhido para o cálculo das curvaturas é o utilizado em [Besl e Jain, 1988], já apresentado na seção 3.1.1. Esta escolha foi motivada pela simplicidade do método, baseado em operadores implementados por máscaras, e pelo fato de ele já ter sido aplicado a dados de profundidade e de intensidade luminosa.

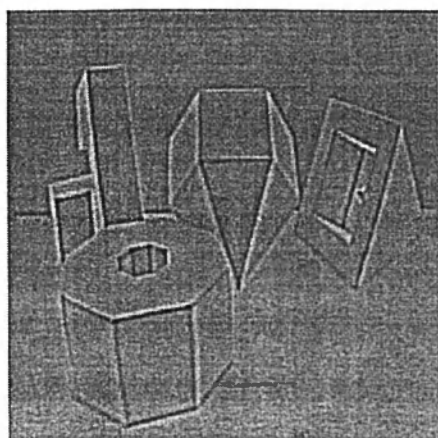
As máscaras das equações 3.6 (página 31) são aplicadas à imagem suavizada através de operações de convoluções, gerando cinco imagens correspondentes aos valores das cinco derivadas parciais. Os valores destas imagens são então utilizados nas equações 3.4 (página 30) para calcular os valores das curvaturas H e K, os quais são armazenados nas duas imagens de saída. As figuras 3.7 e 3.8 contém os pares de imagens de curvaturas obtidos a partir das imagens de entrada da figura 3.3 (página 36).

Nota-se que nas imagens das curvaturas H (figuras 3.7(a) e 3.8(a)) pontos próximos às bordas dos objetos aparecem em destaque. Nos valores de curvatura H da imagem de profundidade (figura 3.8(a)), as bordas de profundidade aparecem mais evidenciadas. Já nos valores de curvatura H da imagem de intensidade luminosa (figura 3.7(a)), essas bordas estão pouco evidenciadas em alguns pontos, mas as bordas correspondentes às bordas de orientação da imagem de profundidade estão bem realçadas.

Observa-se também que nas imagens das curvaturas K (figuras 3.7(b) e 3.8(b)) aparecem realçados pontos de inflexão correspondentes a vértices dos objetos, que são pontos de máximos locais nos valores de curvatura K. Estes pontos de inflexão podem ser utilizados num processo de detecção das bordas entre tais vértices, como por exemplo métodos de detecção de linhas utilizando a transformada Hough [Sonka *et al.*, 1998]. Interessantemente, os pontos de inflexão de bordas de profundidade aparecem mais realçados nas curvaturas K da imagem de profundidade, enquanto aqueles para as bordas de orientação são melhor visualizados nas curvaturas K da imagem de intensidade

luminosa.

Estes fatos acima sugerem a natureza complementar das imagens de intensidade luminosa e de profundidade e a utilidade de sua integração no processo de segmentação.

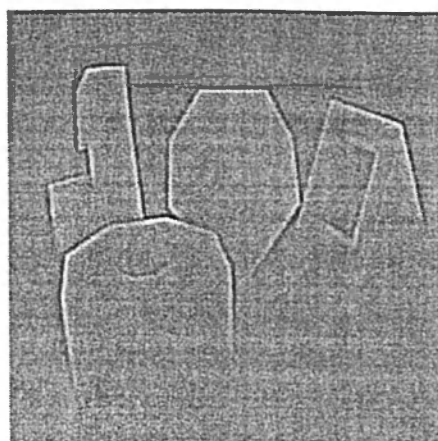


(a) Curvatura H

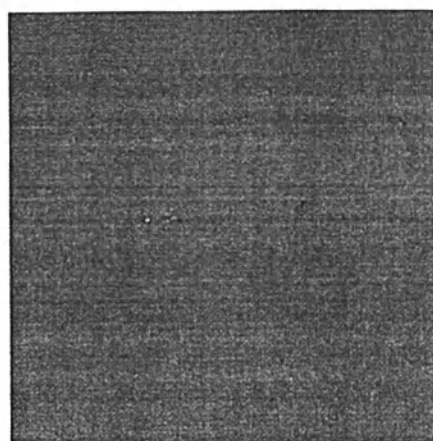


(b) Curvatura K

Figura 3.7: *Imagens de curvaturas obtidas da imagem de intensidade luminosa.*



(a) Curvatura H

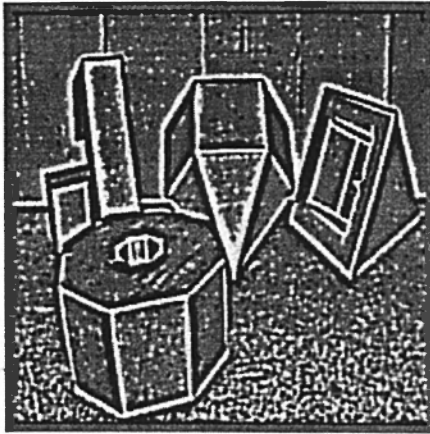


(b) Curvatura K

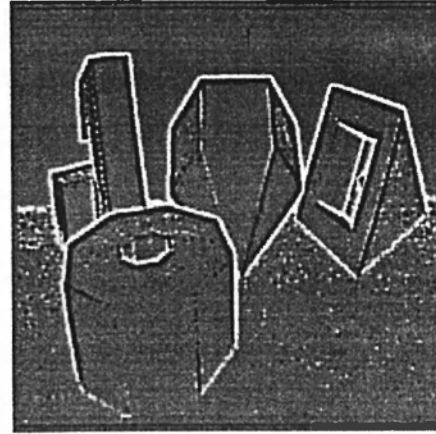
Figura 3.8: *Imagens de curvaturas obtidas da imagem de profundidade.*

3.3.1.3 Cálculo do mapa de sinais HK

Uma vez computadas as imagens de curvaturas H e K, este estágio classifica cada ponto da imagem original como localizado num dos tipos de superfícies ilustrados na figura 3.1 (página 29). Esta classificação é feita analisando-se os sinais dos valores das curvaturas H e K para um mesmo ponto da imagem e, de acordo com a tabela 3.2, atribuindo-se um rótulo de superfície a este ponto (equação 3.3). Os dois parâmetros necessários são os valores de limiar ε_k e ε_h utilizados para determinar o quão próximo de zero os valores das curvaturas serão considerados como tal. Estes limiares são estabelecidos empiricamente.



(a) Mapa para a imagem de intensidade luminosa



(b) Mapa para a imagem de profundidade

Figura 3.9: Mapas de sinais HK para as duas imagens de entrada.

A figura 3.9 mostra os mapas de sinais HK obtidos a partir dos pares de imagens de curvatura obtidos no estágio anterior. Como, neste exemplo, os objetos da cena são poliédricos, os mapas de sinais são constituídos basicamente por apenas 3 dos 8 rótulos de superfície possíveis. Os pontos em preto constituem regiões cilíndricas convexas (*ridges*). Os pontos em branco indicam regiões cilíndricas côncavas (*valleys*) e aqueles em cinza, em maior número, correspondem a regiões planas (*flat*). Alguns poucos pontos próximos aos vértices dos objetos recebem outros rótulos, mas não chegam a formar regiões de tamanho significativo.

Nestes mapas de sinais, podem ser notados os efeitos, nos valores de curvatura, de ruídos ainda remanescentes na imagem mesmo após a suavização. Tais ruídos são responsáveis pelo surgimento de várias pequenas regiões (manchas), claras e escuras, no meio de regiões planas, em cinza. Estas regiões pequenas, que não correspondem a regiões reais dos objetos em cena, são formadas por pontos incorretamente rotulados.

No entanto, a observação mais importante, a qual motivou o desenvolvimento desta abordagem de detecção de bordas, é que muitas das bordas dos objetos estão indicadas, nos mapas de sinais HK, pelas linhas de fronteira entre regiões *ridge* e regiões *valley*, conforme ilustra a figura 3.10(a). A causa disso é o efeito do filtro Gaussiano nas bordas dos objetos. De certo modo, esta suavização também é causada pelas máscaras de aproximação LS utilizadas para calcular as derivadas parciais.

Ainda, foi observado que, em mapas de sinais calculados para imagens de profundidade, estão melhor indicadas bordas de profundidades dos objetos em cena e, por outro lado, suas bordas de orientação estão melhor indicadas em mapas de sinais obtidos para as imagens de intensidade luminosa correspondentes. Em cada par de mapas de sinais obtido, os objetos estão normalmente melhor definidos no mapa da imagem de intensidade luminosa, apesar de o método utilizado para gerá-lo ser mais freqüentemente aplicado apenas a imagens de profundidade. Apesar de que em ambos os mapas de sinais os

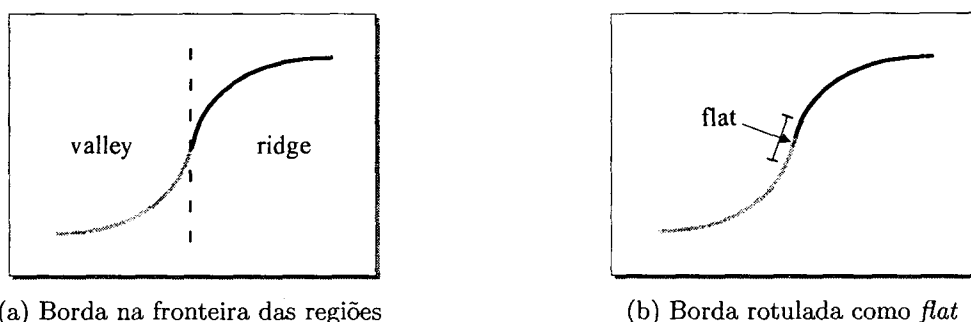


Figura 3.10: Corte transversal de uma borda entre regiões *ridge* e *valley*.

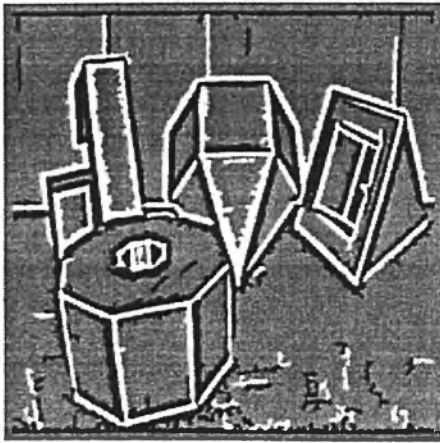
contornos dos objetos aparecerem incompletos, muitas dessas falhas estão localizadas em pontos diferentes, mostrando sua complementaridade e a utilidade da integração aqui realizada e de um modo até então não explorado.

3.3.1.4 Refinamento do mapa de sinais

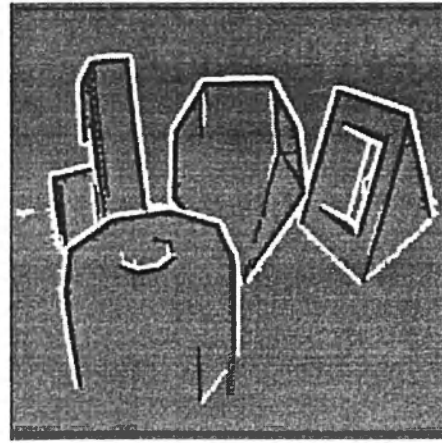
De modo a obter mapas de bordas de melhor qualidade, antes do processo de detecção de bordas, é realizado um processo de refinamento dos mapas de sinais HK obtidos. Um dos objetivos deste refinamento é eliminar regiões muito pequenas, consideradas como produto de ruídos nas imagens de entrada. Além disso, como mostrado na figura 3.10(b), alguns pontos de borda podem ser rotulados, incorretamente, como *flat*. Para melhor preservar a continuidade das bordas a serem detectadas, este estágio de refinamento também é responsável transformar casos como o da figura 3.10(b), deixando-os como na figura 3.10(a).

Para descartar regiões de ruído, realiza-se um crescimento de regiões no mapa de sinais. Este processo agrupa, utilizando vizinhança 4, *pixels* vizinhos que apresentem o mesmo rótulo de superfície, produzindo, assim, um conjunto de regiões de pontos conexos. Regiões com um número de pontos menor que um tamanho mínimo, previamente estipulado, tem seus pontos reclassificados com o rótulo *flat* de modo a não interferir no estágio seguinte, a detecção de bordas. O parâmetro referente ao tamanho mínimo para as regiões serem consideradas é estipulado empiricamente como uma pequena porcentagem do número total de *pixels* na imagem (e.g. 0,03% ou 80 *pixels*). A figura 3.11 mostra os mapas de sinais da figura 3.9 após a reclassificação de regiões ruidosas.

O último passo deste estágio é reclassificar *pixels* rotulados como *flat* em linhas de fronteira entre regiões *ridge* e *valley*. Para isso, é realizado um processo de dilatação de regiões *ridge* e *valley* no qual são reclassificados *pixels* de rótulo *flat* que sejam vizinhos dessas regiões. Nos experimentos realizados, a maior largura de faixas de *pixels flat* sob uma borda foi de dois *pixels*. Deste modo, uma dilatação de apenas um *pixel* para cada região *ridge* e *valley* é considerada suficiente. Quando a largura da faixa *flat* é de apenas um *pixel*, apenas regiões *ridge* são dilatadas (conforme justificado no estágio seguinte). Os resultados da dilatação dos mapas de sinais da figura 3.11 são mostrados na figura



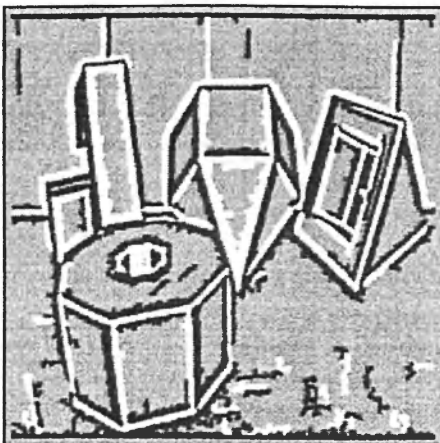
(a) Mapa para a imagem de intensidade luminosa



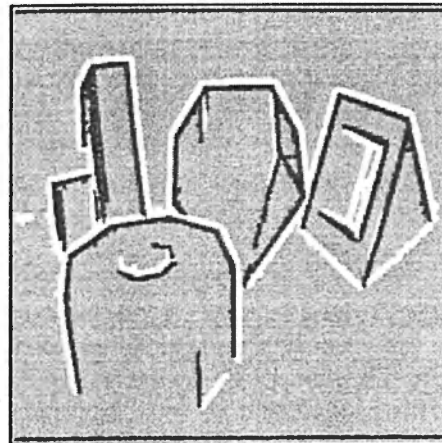
(b) Mapa para a imagem de profundidade

Figura 3.11: Mapas de sinais HK após reclassificação de regiões de ruído.

3.12.



(a) Mapa para a imagem de intensidade luminosa



(b) Mapa para a imagem de profundidade

Figura 3.12: Mapas de sinais HK após dilatação de regiões ridge e valley.

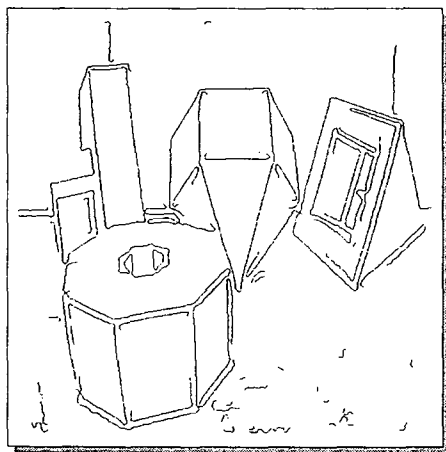
3.3.1.5 Detecção de bordas

Na imagem de entrada deste estágio, o mapa de sinais refinado, as linhas de borda entre as regiões *ridge* e *valley* estão bem definidas, ou seja, sem pontos classificados como *flat*.

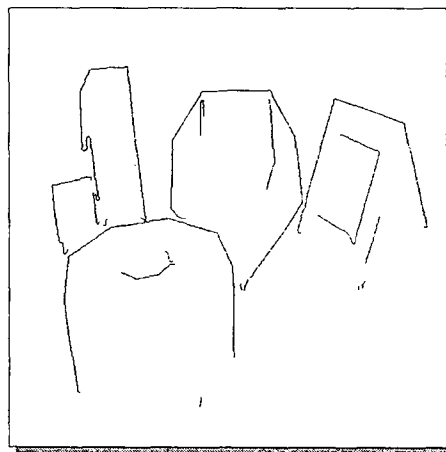
Conforme observado nos experimentos realizados, as bordas dos objetos em cena estão localizadas entre pontos *ridge* e *valley* vizinhos, ao longo das linhas de fronteira entre essas regiões. Assim, para se obter uma borda com largura de apenas um ponto, é necessário escolher se ela será constituída pelos pontos da região *ridge*, de um lado da borda verdadeira, ou pelos pontos da região *valley*, no outro lado da borda. No caso de bordas de profundidade, foi verificado que, nessa linha de fronteira, os pontos da região *ridge* são pontos de objetos, enquanto os pontos classificados como *valley* são

pontos localizados na região de fundo da imagem. Como bordas são consideradas uma propriedade de objetos, e não da região de fundo da imagem, os pontos escolhidos para constituir as bordas são aqueles pertencentes às regiões *ridges* – por isso estas regiões foram priorizadas no processo de dilatação do estágio anterior.

A detecção de bordas consiste, então, em identificar pontos de regiões *ridge* imediatamente adjacentes a pontos de regiões *valley* e atribuir-lhes um rótulo de borda. Os mapas de bordas resultantes dos mapas de sinais da figura 3.12 são mostrados na figura 3.13.



(a) Bordas para a imagem de intensidade luminosa



(b) Bordas para a imagem de profundidade

Figura 3.13: Mapas de bordas extraídos dos mapas de sinais HK.

O método de classificação de superfícies baseado em curvaturas, visto na seção 3.1, pode ser aplicado a qualquer conjunto de dados apresentando a propriedade de coerência de superfície. Porém, a interpretação dos resultados obtidos depende do tipo de dados empregado. Os valores atribuídos pelo sensor para pontos na imagem de intensidade luminosa são de natureza diferente dos valores atribuídos a pontos na imagem de profundidade. Assim, faz-se uma consideração em relação às “bordas de orientação” nas imagens de intensidade luminosa empregadas. Tais bordas encontram-se, em sua maioria, entre superfícies com valores de intensidade luminosa consideravelmente diferentes. Por isso, estas bordas, semelhantes a bordas em forma de degrau, são indicadas da mesma forma que as bordas de profundidade nas imagens de profundidade.

3.3.1.6 Integração dos mapas de bordas

Como as bordas extraídas de ambos os mapas de sinais estão bem localizadas, a integração é feita simplesmente sobrepondo-se os mapas de bordas obtidos no estágio anterior. O resultado desta sobreposição é mostrado na figura 3.14.

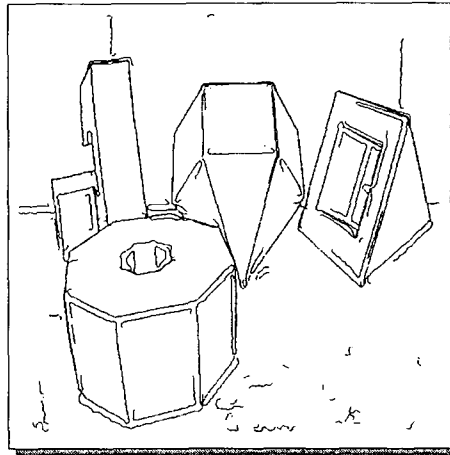


Figura 3.14: Mapa de bordas resultante da integração.

3.3.2 Detecção de Bordas Utilizando a Curvatura H

Em relação aos três rótulos obtidos nos mapas de sinais HK durante os experimentos, foi observado que, na tabela 3.2 (página 29), esses três rótulos pertencem a uma mesma coluna, na qual $K = 0$. Considerando esta coluna de rótulos e analisando o método de extração de bordas descrito anteriormente, nota-se que o fato de as bordas serem indicadas pelas linhas de fronteira entre regiões *ridge* ($H < 0$) e *valley* ($H > 0$) é explicado pela mudança de sinal (*zero-crossing*) que pode ocorrer nos valores da curvatura H nas bordas dos objetos. Os casos nos quais os pontos de bordas são incorretamente rotulados como *flat* ($H = 0$), como na figura 3.10(b), são originados do uso do limiar ε_H em regiões de mudança de sinal da curvatura H.

Ainda, segundo [Besl e Jain, 1988], os zeros da curvatura H formam um bom delineamento das formas dos objetos, semelhantemente a uma imagem – obtida com um operador Laplaciano [Sonka *et al.*, 1998], tipicamente usado em detecção de bordas – dos pontos onde os valores das derivadas passam pelo valor zero (*zero-crossings*).

Devido a essas observações, foram realizados experimentos que originaram um novo método de detecção de bordas baseado apenas nos valores da curvatura H, não sendo necessário, portanto, o cálculo da curvatura K. Esta abordagem de detecção de bordas a partir dos pontos de mudança de sinal da curvatura H foi originalmente utilizada em [Fan *et al.*, 1987], mas não foi aplicada a um par de imagens representando uma mesma cena, como realizado aqui.

A figura 3.15 contém o diagrama de estágios de processamento do método de detecção de bordas apresentado nesta seção. O primeiro estágio executa o mesmo processo de suavização descrito na seção 3.3.1.1. No segundo estágio, as cinco imagens de derivadas parciais e a imagem de valores da curvatura H são calculadas como na seção 3.3.1.2. O próximo estágio, a detecção de bordas a partir dos valores da curvatura H, é descrito a seguir.

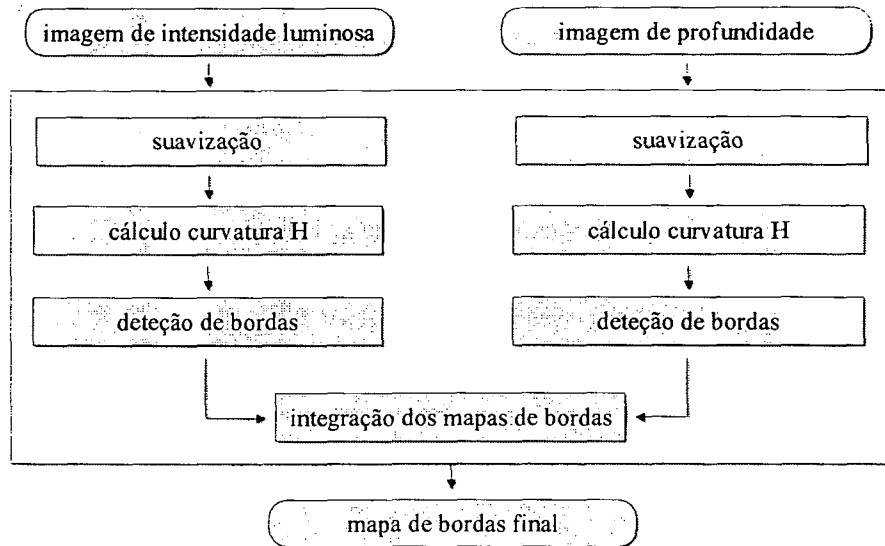


Figura 3.15: Diagrama de estágios de processamento do método de detecção de bordas.

3.3.2.1 Detecção de bordas

Neste estágio, cada ponto da imagem de valores de curvatura H é analisado empregando uma máscara de dimensões 2×2 . O sinal do valor de curvatura do ponto no canto superior esquerdo da máscara é comparado com os sinais dos valores de curvatura dos três pontos vizinhos. Caso o valor de um destes vizinhos apresente sinal diferente do sinal do ponto no canto superior esquerdo, este último é rotulado como ponto de *zero-crossing*.

No entanto, como mostra a figura 3.16, em superfícies ruidosas nem todo ponto onde ocorre uma inversão de sinal nos valores de curvatura pode ser considerado como ponto de borda. Por isso, utiliza-se como suporte os valores de magnitude do vetor gradiente de superfície em cada ponto da imagem,

$$|\nabla f(x, y)| = \sqrt{f_x(x, y)^2 + f_y(x, y)^2} \quad (3.11)$$

onde f_x e f_y são duas das imagens de derivadas parciais obtidas, anteriormente, para o cálculo da curvatura H . Assim, pontos de *zero-crossing* dos valores de curvatura H que possuam magnitude de gradiente maior que um valor de limiar l_g são considerados como ponto de borda.

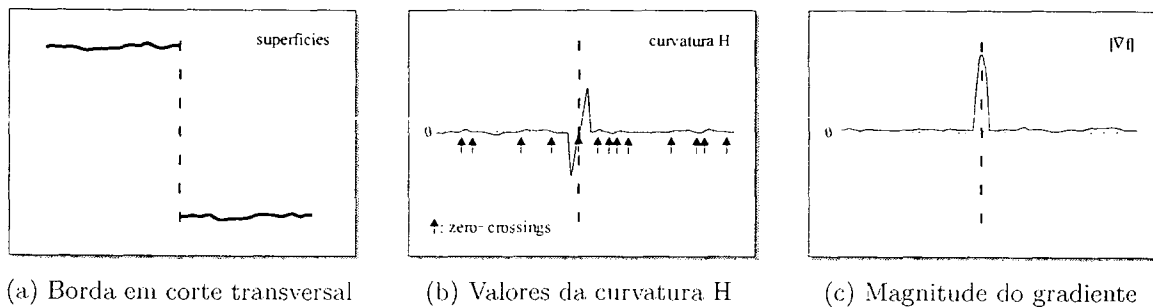
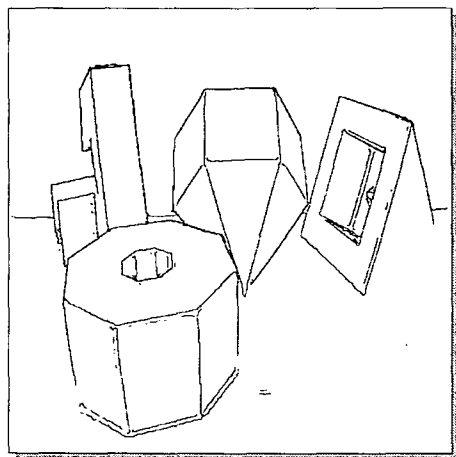
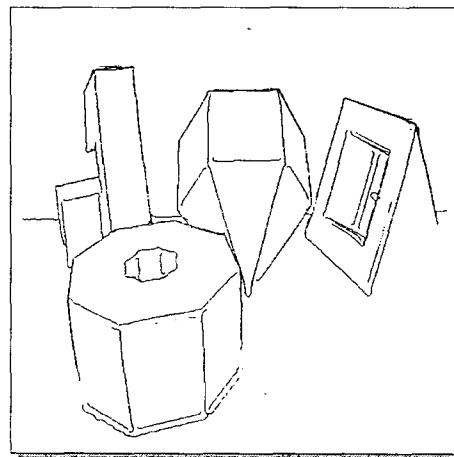


Figura 3.16: Detecção de bordas utilizando zero-crossings da curvatura H .

A figura 3.17(a) mostra o mapa de bordas resultante da identificação destes pontos nos valores de curvatura H calculados para as duas imagens de entrada da figura 3.3. Se a informação de gradiente também for utilizada como suporte na detecção de bordas a partir dos mapas de sinais HK (figuras 3.13 e 3.14), melhores resultados podem ser obtidos para este método, como mostra a figura 3.17(b). As melhorias consistem na eliminação de falsas bordas detectadas em regiões ruidosas, como a região correspondente ao chão da imagem, e nas formas das extremidades das bordas, que ficaram menos arredondadas.



(a) Mapa de bordas detectadas a partir dos valores de curvatura H



(b) Mapa de bordas detectadas a partir dos mapas de sinais HK

Figura 3.17: Mapas de bordas detectadas a partir dos valores de curvatura utilizando informação de magnitude de gradiente.

3.3.3 Resultados Experimentais

Esta seção descreve os experimentos realizados e analisa os resultados obtidos com os dois métodos de detecção de bordas desenvolvidos. Uma comparação visual com os mapas de bordas gerados por outros métodos de segmentação também é realizada.

O ambiente utilizado para desenvolvimento e teste dos métodos propostos foi o pacote KHOROS, um conjunto de ferramentas de programação com predominância das linguagens C e C++ e cujo código fonte é disponibilizado gratuitamente, para fins acadêmicos, na Internet (<http://www.khoros.com>).

Foram realizados testes utilizando a base de imagens Perceptron, disponível em <http://marathon.csee.usf.edu/range/seg-comp/SegComp.html> e também utilizada no trabalho de comparação de métodos de segmentação de Hoover *et al.* [Hoover *et al.*, 1996]. Esta base de imagens é constituída por 40 pares de imagens de intensidade luminosa e de profundidade, sendo que cada par de imagens corresponde a uma mesma cena e possui, também disponível, uma imagem de segmentação gerada manualmente (*ground truth*). Apesar de as cenas das imagens conterem apenas superfícies planas e objetos poliédricos, conforme constatado em Hoover *et al.*, ainda

existem problemas que permanecem sem solução na segmentação destas imagens. Esta afirmação, segundo nosso conhecimento, ainda não foi revogada e os métodos utilizados na comparação permanecem como a principal referência [Jiang *et al.*, 2000].

Diferentemente da maioria dos sensores de profundidade, os quais estão limitados a gerar imagens apenas para objetos pequenos, de dimensões iguais a algumas polegadas, o sensor Perceptron é capaz de fornecer imagens para cenas contendo objetos com dimensões maiores, de alguns metros, o que pode viabilizar um maior número de aplicações (e.g. navegação autônoma). No entanto, as imagens apresentam um nível elevado de ruído, o que, por outro lado, acaba sendo interessante para se analisar o comportamento de métodos de segmentação.

Os mapas de bordas resultantes da aplicação dos métodos desenvolvidos a imagens Perceptron, apresentados a seguir, foram obtidos com a mesma configuração dos parâmetros requeridos pelos métodos. Em ambos, a suavização foi realizada primeiramente com uma máscara de filtro mediano de dimensões 3×3 e, em seguida, a máscara de filtro Gaussiano de dimensões 7×7 , mostrada na equação 3.10. Os demais parâmetros foram configurados empiricamente, utilizando as imagens da base Perceptron, com os seguintes valores:

- Para o método de detecção de bordas a partir de mapas de sinais HK: $\varepsilon_H = \varepsilon_K = 0,001$; limiar de tamanho para regiões de ruídos, a serem descartadas, igual a 0,03% do número total de *pixels* na imagem; número de iterações do processo de dilatação das regiões *ridge* e *valley* igual a 1; e limiar de magnitude de gradiente $l_G = 0.03$.
- Para o método de detecção de bordas a partir dos *zero-crossings* da curvatura H, o único valor de limiar necessário é o limiar de magnitude de gradiente, cujo valor adequado foi estabelecido como $l_G = 0.03$.

Como os experimentos não contemplaram a realização de fechamento dos mapas de bordas obtidos, estes não apresentam contornos fechados para todas as superfícies dos objetos em cena e, assim, não representam segmentações completas das imagens de entrada utilizadas. Por isso, uma avaliação quantitativa utilizando as métricas de [Hoover *et al.*, 1996] não é aplicável a estes resultados. Para realizar uma comparação visual e analisar comparativamente o desempenho dos métodos desenvolvidos quanto à localização correta das bordas detectadas e quanto à preservação das formas dos objetos, os resultados obtidos são mostrados juntamente com a representação por bordas das segmentações geradas pelos quatro métodos do trabalho de Hoover *et al.*.

As figuras 3.18–3.26 mostram, cada uma, as seguintes imagens correspondentes a uma mesma cena: (a) imagem de intensidade luminosa, (b) imagem de profundidade, (c) segmentação manual; (d–g) representação por bordas das segmentações obtidas pelos métodos, em [Hoover *et al.*, 1996], (d) *University of South Florida* (USF), (e) *Washington State University* (WSU), (f) *Universidade of Bern* (UB), Suíça, (g) *University of Edinburgh* (UE), Escócia. Além dos mapas de bordas obtidos com os métodos

propostos, a partir de (h) mapas de bordas HK (UFPR-HK), e (i) *zero-crossings* da curvatura H (UFPR-ZH).

A figura 3.19 mostra resultados para uma cena simples, contendo apenas um objeto. Já as demais figuras, 3.18 e 3.20–3.26 mostram resultados para cenas mais complexas, contendo vários objetos.

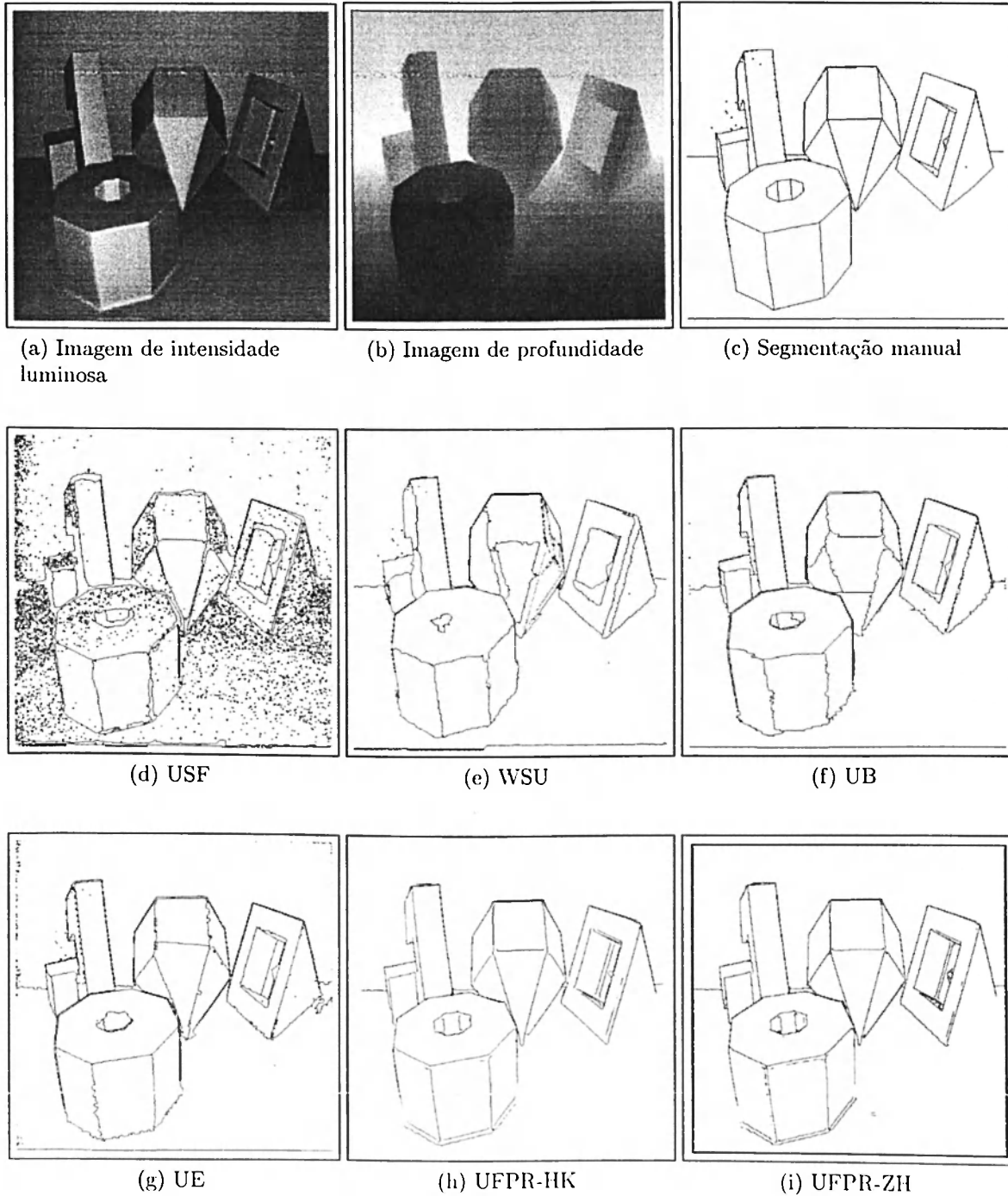


Figura 3.18: Mapas de bordas para imagem de testes Perceptron número 26.

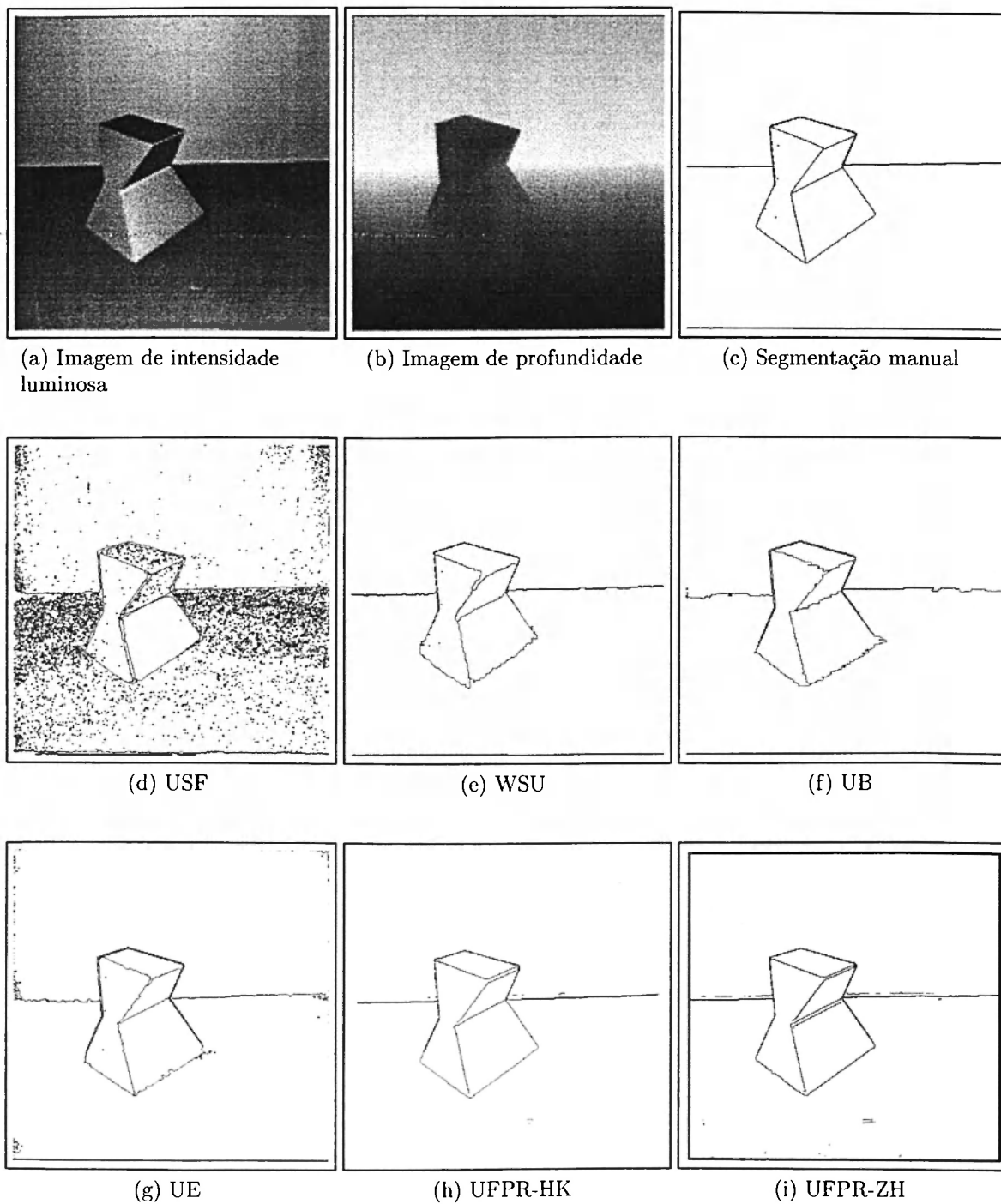


Figura 3.19: Mapas de bordas para imagem de testes Perceptron número 0.

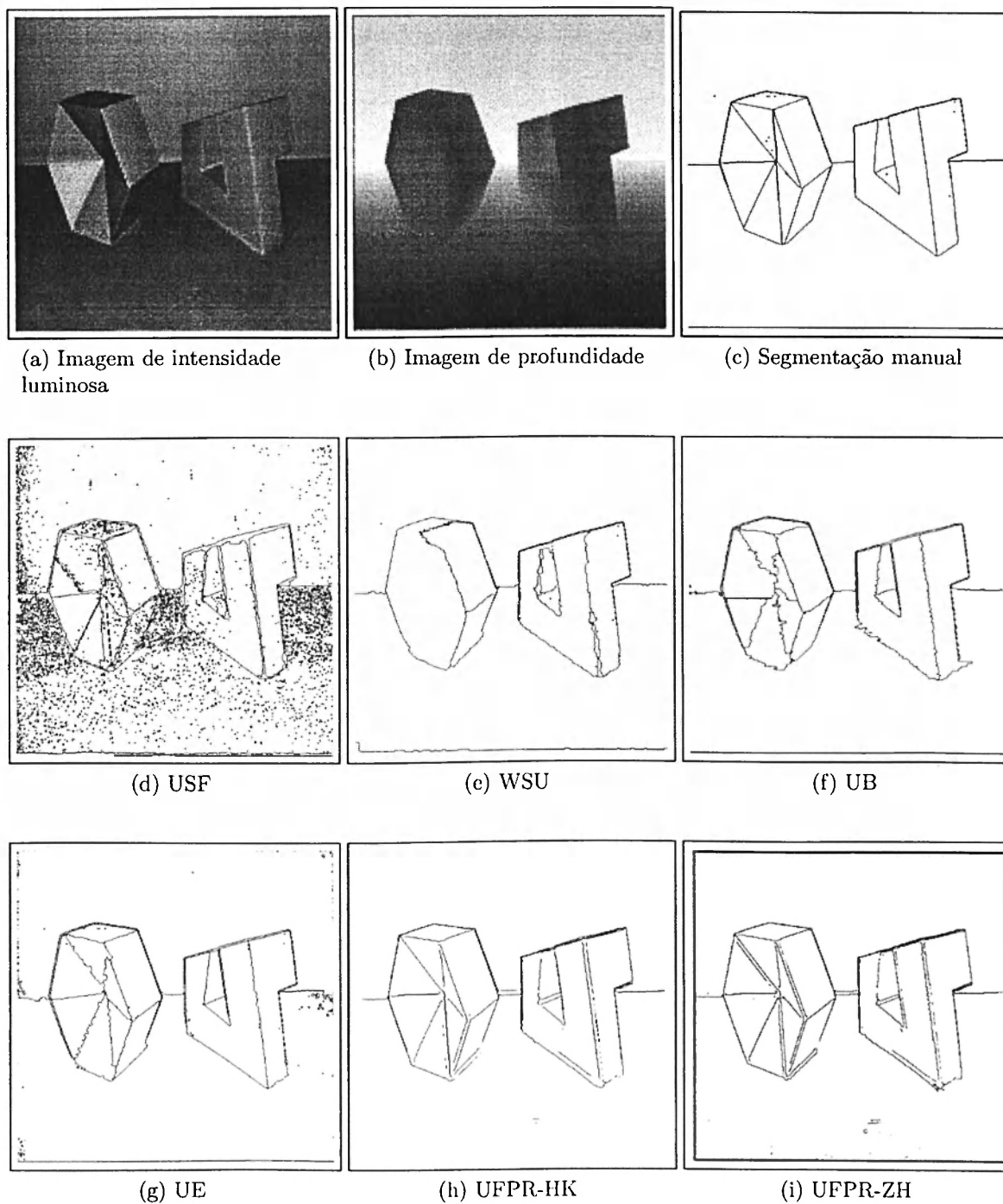


Figura 3.20: Mapas de bordas para imagem de testes Perceptron número 18.

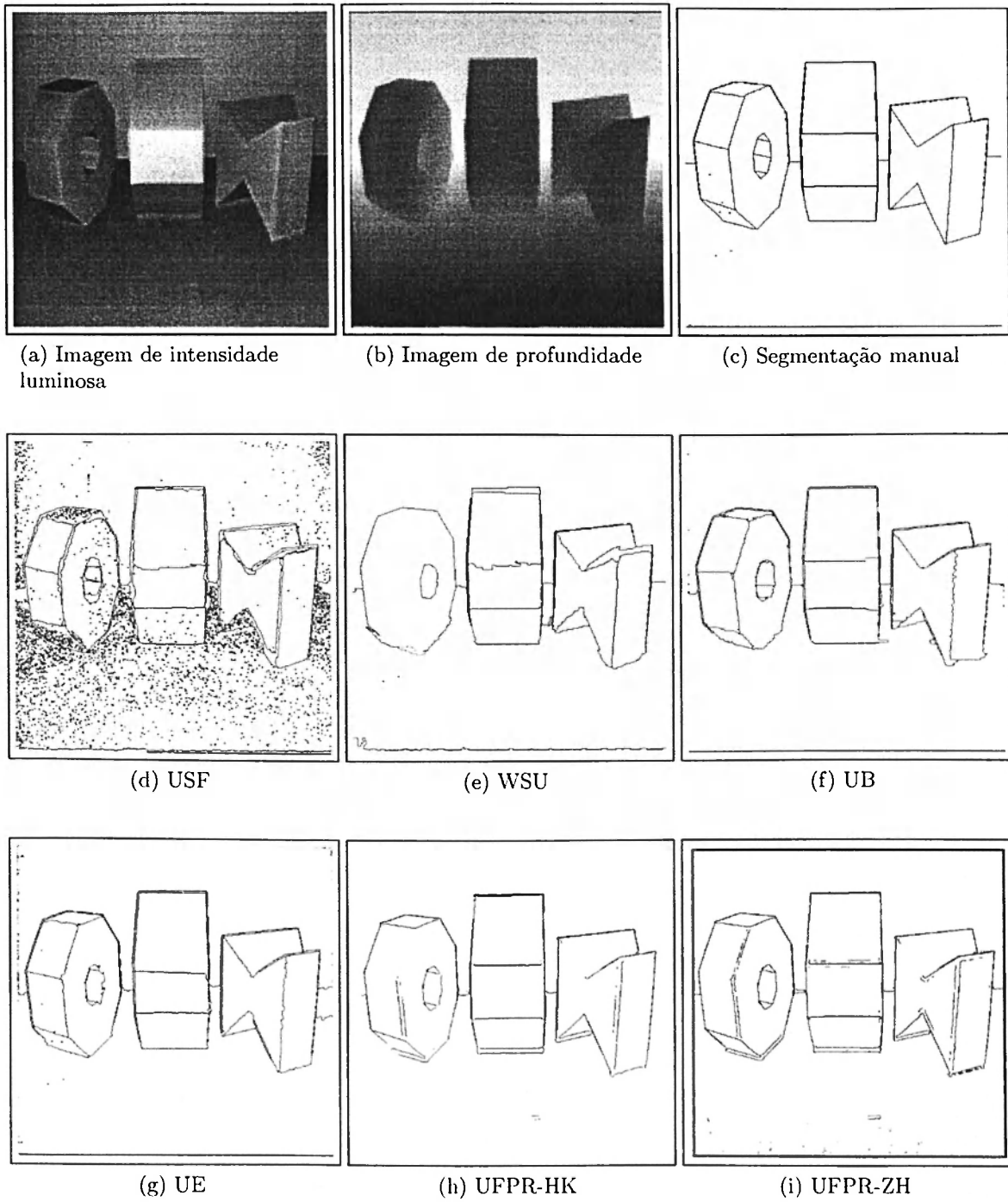


Figura 3.21: Mapas de bordas para imagem de testes Perceptron número 20.

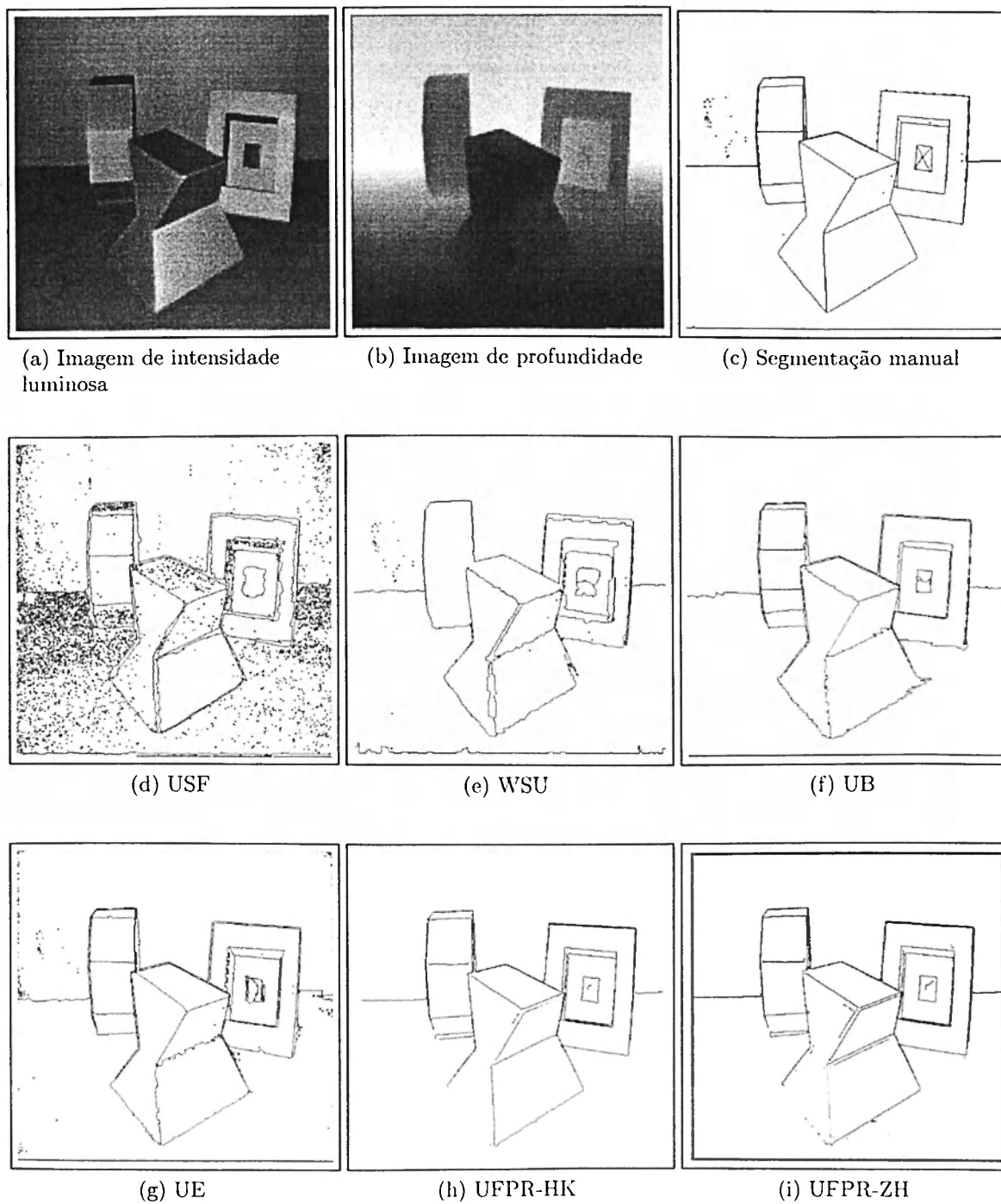


Figura 3.22: Mapas de bordas para imagem de testes Perceptron número 22.

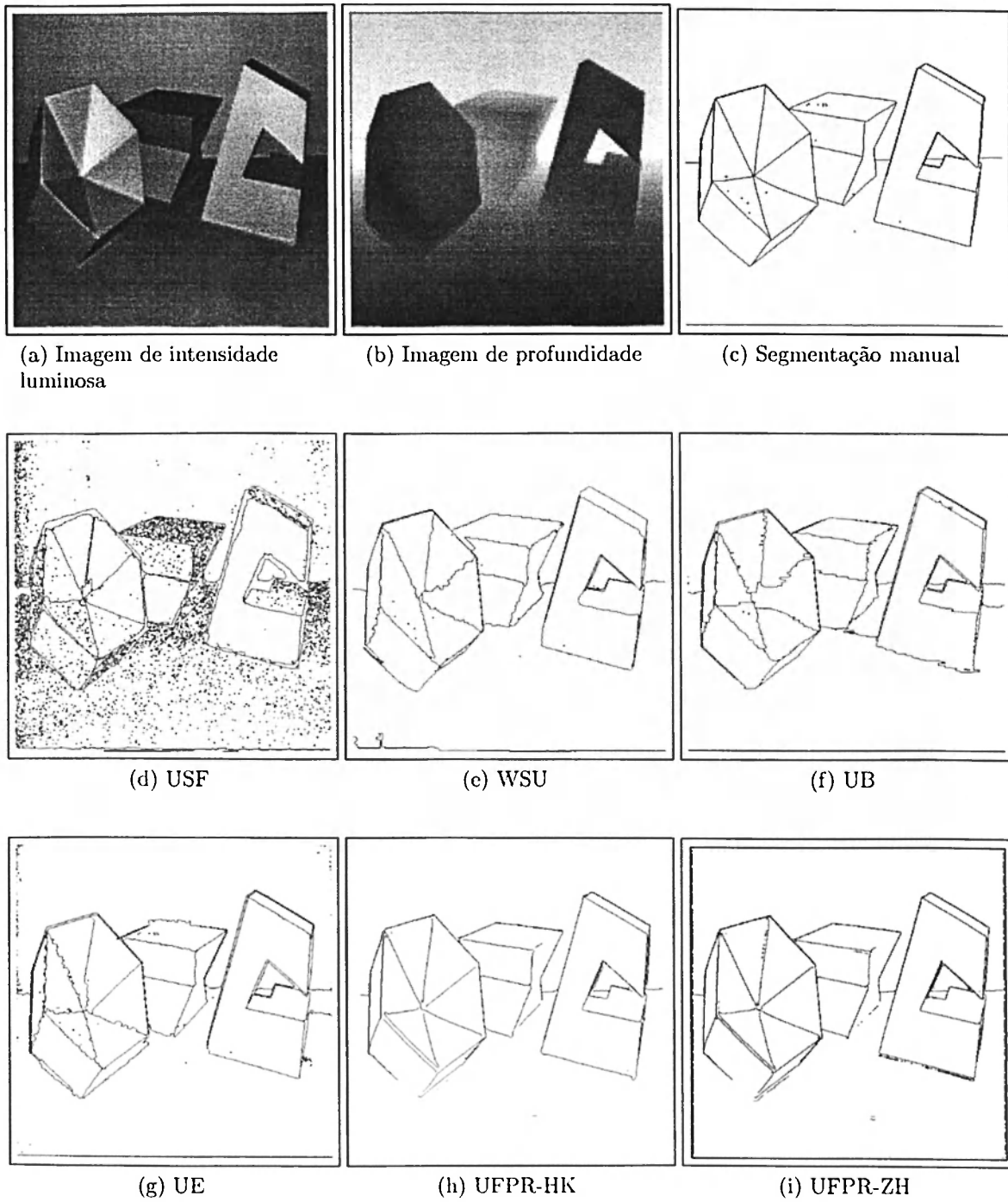


Figura 3.23: Mapas de bordas para imagem de testes Perceptron número 25.

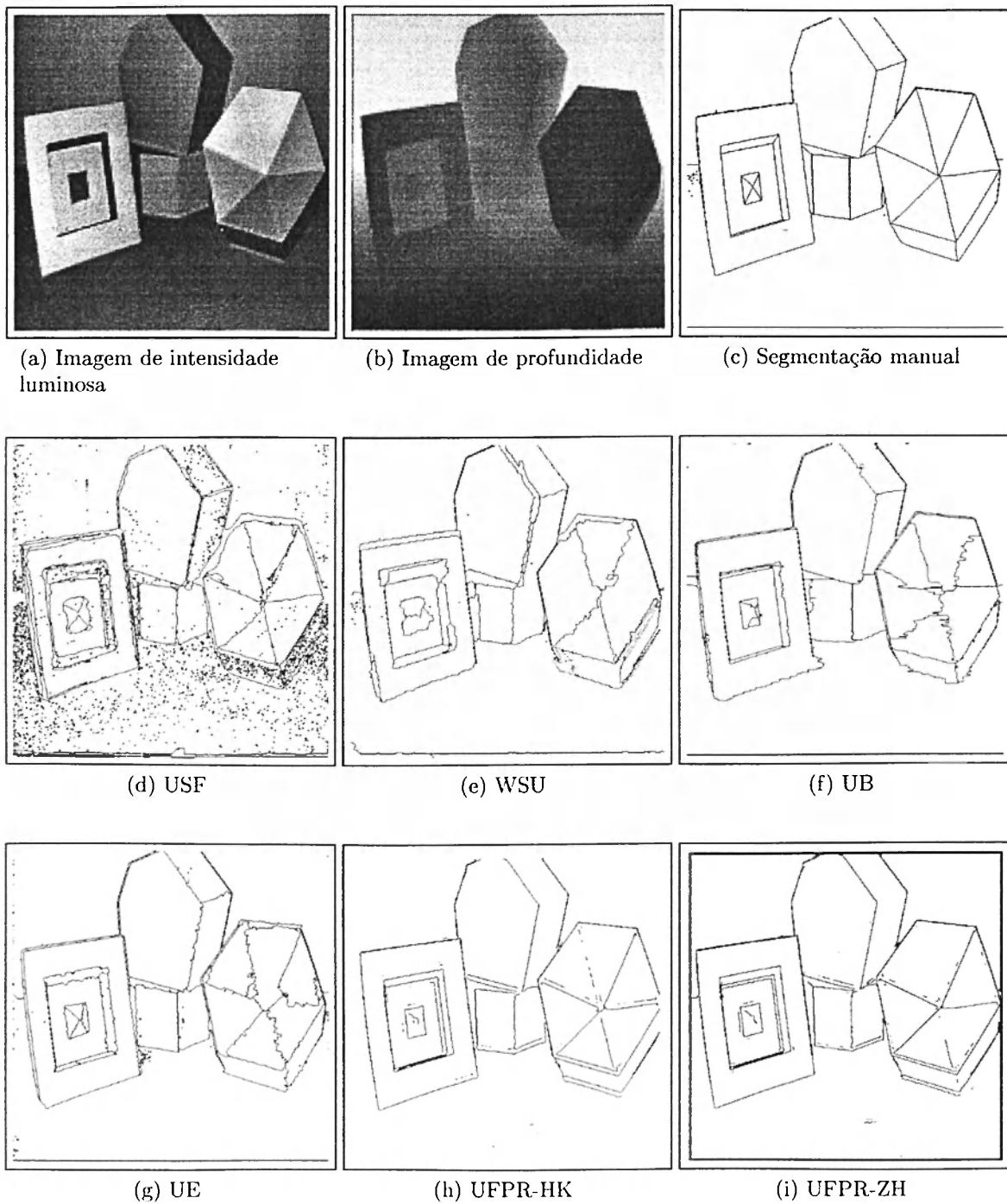


Figura 3.24: Mapas de bordas para imagem de testes Perceptron número 27.

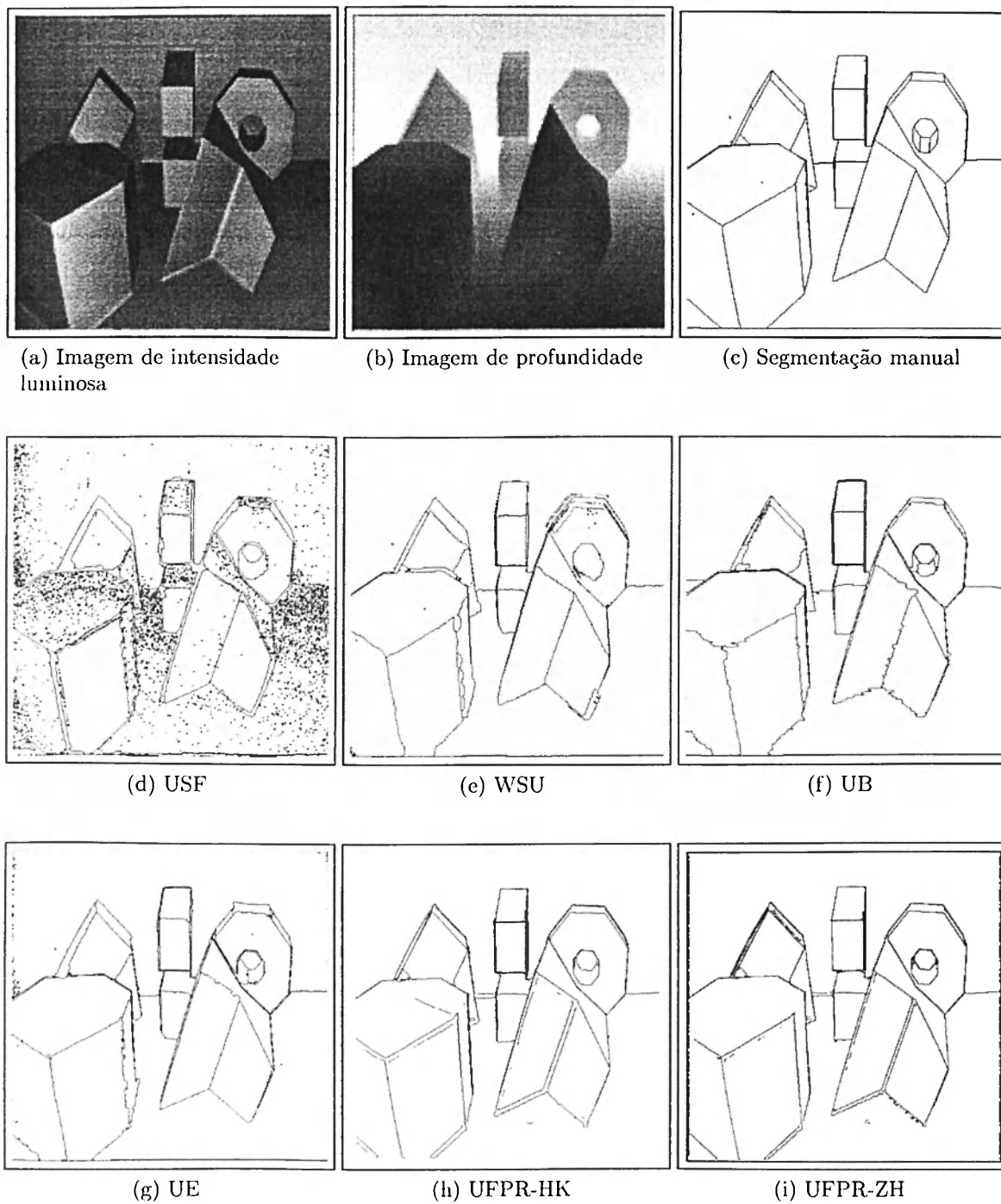


Figura 3.25: Mapas de bordas para imagem de testes Perceptron número 28.

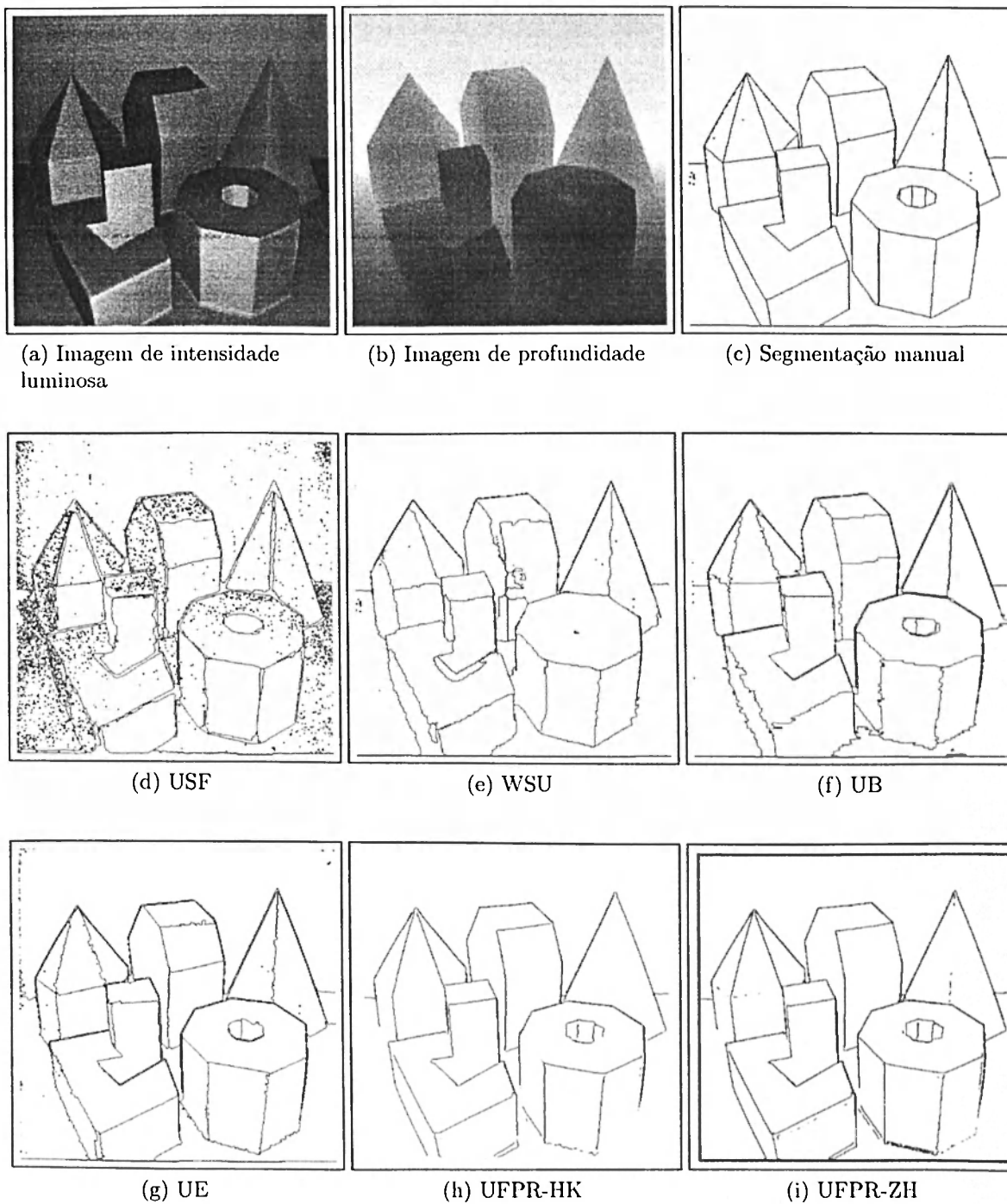


Figura 3.26: Mapas de bordas para imagem de testes Perceptron número 29.

Com base nestes resultados experimentais, pode-se ressaltar alguns pontos positivos e negativos de cada método. Pode-se notar que nenhum dos métodos teve um desempenho considerado perfeito, ou mesmo próximo de 100% de acerto, na segmentação das imagens. Isso já era esperado, já que o problema da segmentação de imagens de profundidade não está resolvido para imagens contendo objetos poliédricos e, tampouco, para aquelas contendo superfícies curvas.

O esquema de avaliação e comparação de algoritmos de segmentação de imagens de profundidade proposto por Hoover *et al.* constitui uma importante contribuição à área de pesquisa em visão computacional porque permite analisar as limitações dos métodos desenvolvidos. Como os resultados mostram, o algoritmo USF é extremamente sensível a ruídos e não fornece uma segmentação com regiões corretamente delimitadas. O algoritmo WSU é o mais lento e não apresentou bons resultados com relação à preservação da topologia de objetos e à sensibilidade a ruídos. O método UB é muito rápido, mas apresentou grandes perdas com relação à localização precisa das bordas entre as regiões, dificultando um processo de representação dessas imagens. Já o algoritmo UE apresentou bons resultados com relação à descrição topológica das superfícies. Porém, este também é sensível a ruídos e há a distorção da forma de algumas bordas. Todos apresentaram problemas, principalmente, na correta segmentação de regiões pequenas das imagens e na preservação das formas das faces dos objetos. Devido ao alto nível de ruído das imagens Perceptron, até mesmo regiões relativamente grandes dessas imagens foram incorretamente segmentadas.

Por causa da contribuição da imagem de intensidade luminosa, os métodos HK e ZH mostraram-se menos sensíveis a ruídos e, apesar de ser um algoritmo relativamente simples, as bordas detectadas apresentam uma boa localização e melhor preservam a forma dos objetos. Os métodos detectaram também faixas (marcas) próximas às bordas (bordas de intensidade) dos objetos, visíveis nas imagens de intensidade luminosa da base Perceptron, que não correspondem a bordas reais. Visualmente, os mapas de bordas fornecidos pelos dois métodos desenvolvidos são muito parecidos, sendo que os resultados produzidos pelo método HK são ligeiramente melhores que aqueles fornecidos pelo método ZH por apresentarem um menor número de pontos de bordas falsos (bordas duplas). No entanto, o método HK utiliza um maior conjunto de informações e apresenta maior custo computacional que o método ZH, o qual não requer os cálculos de curvaturas K e de mapas de sinais HK – portanto, eliminando também a necessidade de se especificar dois valores de limiar, ε_H e ε_K , e de refinar os mapas de sinais.

Uma limitação dos métodos desenvolvidos é que eles não foram capazes de detectar algumas das bordas dos objetos. Estes casos ocorrem para bordas onde são pequenas as diferenças entre os valores de intensidade luminosa ou profundidade entre pontos nos diferentes lados da borda. Para detectar estas bordas, o método ZH pode ser expandido de modo a identificar também pontos onde ocorrem máximos locais nos valores de curvaturas H , identificando bordas como mostrado na figura 3.2. Porém, a correta identificação de

máximos locais requer um cálculo mais preciso dos valores de curvatura H , pois os ruídos presentes nas imagens de entrada fazem com que apareçam vários pontos com valores extremos de curvatura e que não estão sobre uma borda. Porém, o desenvolvimento de métodos de cálculo de curvaturas mais robustos aos efeitos de ruídos ainda é um desafio na área de pesquisa em visão computacional. Ainda, para eliminar os falsos pontos de borda e realizar um fechamento do mapa de bordas, faz-se necessário o desenvolvimento de métodos de refinamento para serem aplicados aos resultados obtidos.

Apesar de requerer um par de imagens como dados de entrada, os métodos de segmentação desenvolvidos neste trabalho apresentaram tempos de processamento relativamente baixos, devido à sua simplicidade, mas que não podem ser diretamente comparados com os tempos dos algoritmos USF, WSU, UB e UE, executados em equipamentos diferentes. O tempo médio de obtenção do mapa de bordas final foi de 4 segundos para ambos os métodos, HK e ZH, executados em um computador PC com processador Pentium IV 2 GHz, 512 MB de memória, e com o sistema operacional Linux versão 2.4.18. Se os métodos propostos forem integrados a algoritmos de segmentação que já utilizem informações das curvaturas H e K , então o custo computacional adicionado pelos métodos de detecção de bordas será consideravelmente pequeno pois o tempo dominante do processo consiste no cálculo das curvaturas. Uma outra observação é que, caso arquiteturas computacionais paralelas sejam utilizadas, os métodos HK e ZH possibilitam a obtenção simultânea do mapa de bordas intermediário para a imagem de intensidade luminosa e daquele para a imagem de profundidade. Essas considerações sobre custo computacional e paralelismo são importantes para aplicações do método em sistemas cujos tempo de resposta deve ser pequeno, como as aplicações em tempo real (e.g. navegação autônoma).

Uma possível aplicação dos métodos desenvolvidos seria sua integração a algoritmos de segmentação por regiões como o algoritmo UE, baseado no trabalho de Trucco e Fisher e que se utiliza de informações de curvaturas de superfícies, formando um algoritmo híbrido e com capacidade de melhor localizar as bordas e preservar as formas dos objetos no resultado da segmentação.

3.4 Conclusão

A segmentação de imagens ainda constitui um dos maiores desafios na área de pesquisa em visão computacional porque deve identificar, com precisão, as formas e a localização das bordas dos objetos em cena. Para isso, este processo depende fortemente da correta extração de características dos objetos, a partir de imagens contendo ruídos.

Na caracterização e segmentação de imagens, as medidas de curvaturas de superfície vêm sendo muito utilizadas porque são invariantes a transformações geométricas sólidas e agregam informações úteis à identificação das superfícies digitais contidas nas imagens. Na maioria das vezes, estas curvaturas têm sido calculadas a partir de imagens de

profundidade. No entanto, bons resultados também haviam sido obtidos na caracterização de imagens de intensidade luminosa, sem que esforços de integração destes dois tipos de imagens para fins de segmentação tivessem sido realizados.

A maior parte das pesquisas em visão computacional esteve limitada ao uso de apenas um tipo de imagem como informação de entrada. Contudo, a vantagem de se utilizar mais de um sensor é clara da observação de que diferentes sensores são sensíveis a diferentes sinais, cada qual revela um conjunto particular de propriedades do ambiente observado. Acredita-se que a integração de dados diferentes em seus princípios pode fornecer uma maior confiabilidade e flexibilidade a métodos de segmentação de imagens, diminuindo as restrições comumente encontradas quando da utilização de apenas um tipo de dado em separado, bem como eliminando lacunas de informações e casos de difícil tratamento.

Para superar algumas das limitações dos métodos de segmentação atuais, foram propostos, neste capítulo, dois métodos inéditos de detecção de bordas que podem ser utilizados na segmentação de imagens de profundidade. Estes métodos são baseados em curvaturas de superfícies calculadas de a partir de imagens de intensidade luminosa e de profundidade, correspondentes à mesma cena.

A integração de imagens realizada resulta num maior volume de dados a serem processados, o que, à primeira vista, representaria um custo computacional alto. Contudo, nos experimentos realizados, os processos de detecção de bordas apresentaram custos computacionais baixos devido à sua simplicidade. Os métodos desenvolvidos neste trabalho também mostram-se muito adequados quanto à utilização de processamento em paralelo. Mais importante ainda, como mostrado na seção 3.3, em vários pontos as informações obtidas dos dados de intensidade luminosa e de profundidade são complementares. Os resultados obtidos nos experimentos realizados justificam a integração dos dados e confirmam a viabilidade da abordagem adotada para a segmentação de imagens de profundidade.

Apesar de as imagens da base de dados utilizada, Perceptron, apresentarem um nível de ruído elevado e os mapas de bordas obtidos com os métodos propostos não apresentarem contornos fechados para todas as faces dos objetos, as bordas detectadas mostram-se bem localizadas e preservam a forma dos objetos. Embora consideravelmente simples, o método mostra-se potencialmente adequado à utilização em sistemas de visão, por exemplo, integrando-se-o a outro método de segmentação por curvaturas de superfícies baseado em regiões.

De modo a melhorar os resultados obtidos, um método de cálculo de curvatura mais preciso é necessário. Porém, o desenvolvimento de tal tipo de método é um objetivo ainda perseguido por muitos pesquisadores.

As curvaturas das superfícies são muito sensíveis a ruídos porque requerem o cálculo de derivadas segundas, nas quais os efeitos dos ruídos são amplificados. Outro problema é que a operação de convolução de máscaras quadradas no cálculo de curvatura contribui para que, em alguns casos, a propriedade de coerência de superfície não seja válida. Por

exemplo, quando as máscaras são posicionadas sobre uma borda, são utilizados no cálculo pontos pertencentes a superfícies distintas, produzindo alguns resultados inadequados.

Caso o cálculo das imagens de derivada parcial fosse mais preciso, o problema da não detecção de algumas bordas dos objetos poderia ser solucionado através da detecção de descontinuidades na orientação das superfícies identificadas a partir das imagens de coeficientes dos vetores normais, calculados – com um custo computacional baixo, conforme mostrado na página 31 – a partir das imagens de derivadas parciais, de modo semelhante ao realizado em [Yokoya e Levine, 1989]. Entretanto, experimentos mostraram que as imagens de coeficientes dos vetores normais calculadas pelo método atual não são adequadas.

Para obter melhores estimativas de curvaturas, é preciso melhorar os métodos analíticos atualmente utilizados. Conseqüentemente, faz-se necessário o emprego de métodos de aproximação de superfícies mais eficientes. Um dos principais problemas dos métodos de aproximação de superfícies é que, além de variações aleatórias em pequena escala (ruídos de quantização), as imagens de entrada contêm dados pertencentes a diferentes superfícies, sendo que alguns pontos não podem ser associadas a nenhuma superfície (ruído impulsivo). Assim, para estimar corretamente os parâmetros de uma superfície é necessário descartar os pontos não pertencentes a ela, o que é uma tarefa complexa.

Por isso, alguns experimentos em segmentação de imagens de profundidade foram realizados empregando algoritmos conhecidos como *métodos de estimativa robustos*, os quais são projetados especificamente para, ao estimar os parâmetros de uma superfície, tolerar dados de outras e pontos não obedecendo ao modelo de superfície assumido. Estes experimentos, os quais resultaram num novo método de segmentação de imagens de profundidade, são apresentados no capítulo a seguir.

Capítulo 4

Segmentação por Extração de Superfícies

Os problemas encontrados durante o processo de aproximação de superfícies para o cálculo das curvaturas H e K , relatados nos experimentos do capítulo anterior e causados pela interferência de ruídos e de pontos de diferentes superfícies, fizeram com que os esforços de pesquisa fossem direcionados à análise de métodos mais eficientes de aproximação conhecidos coletivamente como *métodos de estimativa robustos*, ou ainda, *métodos de aproximação robustos*.

Além da aproximação, estes métodos viabilizam a solução do um problema mais complexo, que é a *extração* de superfícies. Este problema consiste não apenas da determinação dos parâmetros de um modelo de superfície, mas também na identificação do conjunto de pontos que constituem tal superfície.

Assim, essa nova direção nas pesquisas realizadas resultou no desenvolvimento de um novo método de segmentação, a princípio, de imagens de profundidade. Este método é baseado em uma versão aperfeiçoada de um método de aproximação robusto, o qual emprega um processo de otimização evolucionária na forma de um algoritmo genético. Uma avaliação quantitativa do método também foi realizada utilizando as métricas e a base de imagens do trabalho de Hoover *et al.* [Hoover *et al.*, 1996].

Na seção 4.1 são discutidos os principais problemas encontrados por métodos de aproximação mais simples, como o LS. Os métodos de aproximação robustos mais conhecidos são descritos na seção 4.2. A seção 4.3 apresenta o problema de extração de primitivas geométricas e descreve como algoritmos genéticos têm sido utilizados para resolvê-lo. Trabalhos relacionados à extração de superfícies para fins de segmentação de imagens de profundidade são descritos na seção 4.4. Por fim, o método de segmentação desenvolvido, bem como seus resultados experimentais e avaliação quantitativa, são apresentados na seção 4.5. Considerações finais são feitas na seção 4.6.

4.1 *Outliers* e o Método *Least Squares*

O método *Least Squares* (LS), visto na seção 2.4 (página 19), é freqüentemente utilizado por métodos de segmentação de imagens para a aproximação de superfícies. No entanto, o LS é extremamente sensível a ruídos porque, no cálculo da melhor aproximação, leva em consideração todos os pontos do conjunto, incluindo *outliers*. Em consequência disso, a aproximação acaba sendo corrompida, ou seja, desviada do resultado correto.

Exemplos de aproximações corrompidas são mostradas nas figuras 4.1(a) e 4.1(b), onde o LS foi utilizado para aproximar uma linha reta a um conjunto de pontos. Como mostra a figura 4.1(c), um único *outlier* pode ser capaz de corromper a aproximação LS, não importando o número de pontos não afetados por ruídos. Estes problemas também ocorrem na aproximação de superfícies e podem ser melhor entendidos caso cada exemplo na figura 4.1 seja considerado como uma visão em corte transversal de pontos medidos para uma superfície plana.

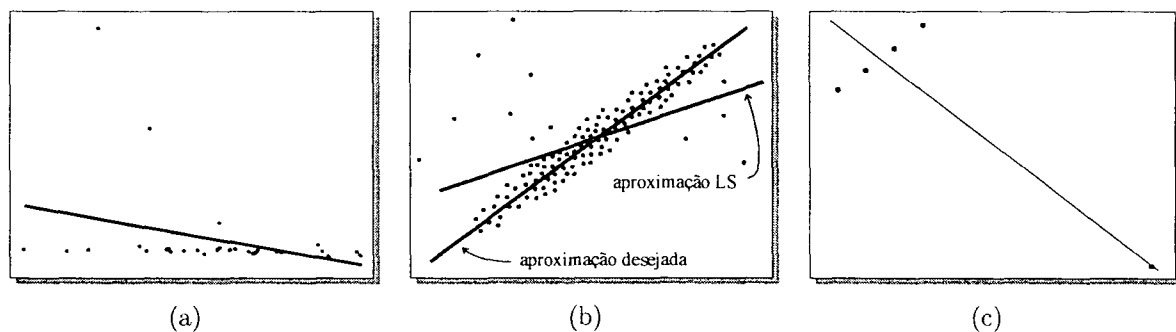
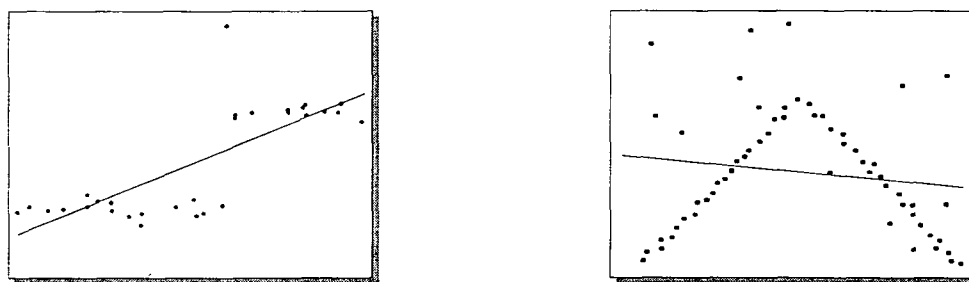


Figura 4.1: Aproximações incorretas do método LS devido à presença de ruídos

Uma outra forma de se entender a natureza dos *outliers* é definindo-os como sendo pontos capazes de introduzir um grande desvio na superfície aproximada em relação à aproximação ideal. Segundo essa definição, nota-se que eles não se constituem somente dos pontos discrepantes de uma imagem, aqueles que não pertencem a nenhuma superfície real da cena (ruído impulsivo). Como mostram as figuras 4.2(a) e 4.2(b), quando uma determinada superfície de um objeto está sendo aproximada, os pontos das outras superfícies, principalmente das superfícies vizinhas (a abordagem de aproximação comumente utilizada é a local), também são capazes de corromper a aproximação. Portanto, esses pontos também devem ser considerados como *outliers*. Mas já que eles não são pontos discrepantes, pois pertencem a uma das outras superfícies (são *inliers* para uma das outras superfícies), eles são denominados *pseudo-outliers* em relação à superfície que está sendo aproximada.

Deste modo, mesmo que os *outliers* sejam completamente removidos através do uso de um filtro de suavização, e mesmo que os efeitos colaterais desse filtro sejam atenuados ou eliminados (o que em muitos casos ainda não foi conseguido), ainda assim existirão *pseudo-outliers*, os quais devem ser tratados adequadamente para que o procedimento de



(a) Região de borda em degrau entre “planos”

(b) Região de borda de orientação entre “planos”

Figura 4.2: Aproximações corrompidas por outliers e pseudo-outliers

aproximação de superfícies não seja corrompido.

Na seção seguinte são apresentados métodos projetados especificamente para lidar com a presença de *outliers* em geral. No restante deste texto, a não ser quando explicitado de outra forma, o termo *outlier* será utilizado genericamente em relação a pontos afetados por ruído impulsivo e a *pseudo-outliers*.

4.2 Métodos de Estimativa Robustos

O termo *robusto*, originário da área de estatística, pode ser definido de várias formas, mas em geral, quando referido a métodos estatísticos para a estimativa de parâmetros (*robust estimators*), é empregado com o seguinte significado [Huber, 1981]:

Insensível a pequenos desvios em relação às condições ideais para as quais o método de estimação é otimizado.

Nessa definição, a expressão “pequenos desvios” pode ter duas interpretações diferentes, ambas importantes: (1) pequenos desvios em todo o conjunto de dados; ou (2) desvios grandes em um número pequeno de elementos desse conjunto. Geralmente, é esta última interpretação, relacionada à idéia de *outliers*, que diz respeito aos maiores problemas encontrados pelos métodos de aproximação de superfícies tradicionais. As principais dificuldades destes métodos estão relacionadas ao fato de que, além de variações aleatórias em pequena escala (ruído de quantização) presentes nos dados, as imagens de entrada quase sempre contêm dados pertencentes a diferentes populações estatísticas, ou seja, diferentes superfícies em cena. Ainda, alguns dos dados podem corresponder a medições que não podem ser associadas a nenhuma dessas populações (ruído impulsivo).

Métodos robustos [Meer *et al.*, 2000] [Stewart, 1999] são vantajosos porque são projetados especificamente para, ao estimar os parâmetros que descrevem a população dominante, trabalhar com dados de múltiplas populações e tolerar pontos não obedecendo ao modelo assumido. Vários dos métodos robustos inicialmente utilizados em visão computacional são métodos de estimação originários da área de estatística. Porém, dados visuais são geralmente mais complexos que os dados tipicamente analisados em estatística.

e, assim, a aplicação direta dessas técnicas robustas frequentemente não produz resultados satisfatórios. Por isso, aprimoramentos e novos métodos robustos foram desenvolvidos na área de visão computacional.

A medida de robustez mais utilizada na literatura para comparar métodos robustos é a de *breakdown point*, ou ponto de colapso: a menor fração de *outliers*, no conjunto de dados, capaz de corromper a aproximação, levando o método a produzir resultados arbitrariamente ruins. Esta é uma medida de pior caso e quanto maior o seu valor, mais tolerante é o método à presença de *outliers*. O método LS, por exemplo, possui *breakdown point* zero (figura 4.1(c)).

A seguir serão apresentados alguns dos métodos robustos mais utilizados em visão computacional e que, assim como o LS (equação 2.1), também se utilizam de uma *função objetivo* na estimação dos parâmetros. Esta função geralmente consiste em minimizar um custo, ou erro, definido em termos dos *valores residuais* para os N pontos do conjunto de dados, definidos por $r_{i,a}$, $1 \leq i \leq N$. Idealmente, cada um destes valores deve ser a verdadeira distância geométrica (Euclidiana) entre um ponto P_i e a superfície determinada pelo vetor de parâmetros a .

4.2.1 Least Median of Squares

Desenvolvido na área de estatística, o método *Least Median of Squares* (LMS) [Rousseeuw, 1984], ou Mínima Mediana de Quadrados, possui *breakdown point* de, exatamente, 0,5. Isto significa que ele obtém bons resultados para aproximações em conjuntos de pontos formados por até 50% de *outliers*.

A mediana de um conjunto de $N + 1$ valores é o valor m para o qual exatamente $\frac{N}{2}$ valores do conjunto são menores ou iguais a m e os outros $\frac{N}{2}$ valores são maiores ou iguais a m . Um modo simples de determinar a mediana é ordenar o conjunto de valores e tomar o valor na posição $\frac{(N+1)}{2}$, o que tem um custo computacional mínimo $O(N \log N)$. No entanto, algumas técnicas de seleção [Press *et al.*, 1992] possibilitam a determinação do n -ésimo maior elemento de um conjunto de dados sem que seja necessária a ordenação de todos os elementos desse conjunto. Estas técnicas podem ser utilizadas para encontrar a mediana em tempo $O(N)$. Quando o conjunto tem um número par de valores, é convenção considerar como mediana a média aritmética entre os dois valores centrais do conjunto ordenado.

Os parâmetros \hat{a} estimados pelo LMS são aqueles que minimizam a mediana dos quadrados dos resíduos. A função objetivo do LMS é mostrada na equação 4.1.

$$\hat{a} = \underset{a}{\operatorname{argmin}} \left\{ \operatorname{median}_i \left\{ r_{i,a}^2 \right\} \right\} \quad (4.1)$$

A intuição por detrás do LMS é que até metade dos pontos podem estar localizados arbitrariamente longe da estimativa desejada sem que o valor da função objetivo seja

alterado. Ou seja, como o objetivo é minimizar a mediana dos resíduos, garante-se que, no final do processo de otimização, $\frac{N}{2}$ resíduos serão ainda menores que a mediana mínima. Os $\frac{N}{2}$ resíduos restantes – dentre estes, possivelmente, os resíduos de *outliers* – serão desconsiderados e não exercerão influência no valor da função objetivo.

Para encontrar o vetor de parâmetros $\hat{\mathbf{a}}$, é necessário minimizar a função *median*. Para isso, utiliza-se uma técnica de otimização desenvolvida independentemente nas áreas de visão computacional [Fischler e Bolles, 1981] e estatística [Rousseeuw, 1984] e denominada *técnica de amostragem aleatória (random sampling technique)*. A idéia consiste em aleatoriamente selecionar um número k de subconjuntos contendo, cada um, n pontos, onde n é o número mínimo de pontos necessários para instanciar o modelo (e.g. dois pontos para uma linha reta, três para um plano ou nove para um polinômio biquadrático). Um vetor de parâmetros \mathbf{a}_{s_j} é então instanciado, por interpolação, para cada subconjunto s_j , com $1 \leq j \leq k$. Cada \mathbf{a}_{s_j} é considerado como uma aproximação hipotética e são calculados os quadrados das distâncias residuais, $r_{i,\mathbf{a}_{s_j}}^2$, para os demais $N - n$ pontos de modo a obter o valor da mediana. A melhor hipótese, o vetor \mathbf{a}_{s_j} correspondente à menor mediana dentre todos os subconjuntos, é escolhida como o mínimo da função objetivo, o vetor de parâmetros $\hat{\mathbf{a}}$. No total, este processo requer tempo¹ $O(kN)$ utilizando técnicas de seleção para encontrar as medianas.

A figura 4.3 contém um exemplo de aplicação do LMS com amostragem aleatória para estimar os parâmetros de uma reta. As várias hipóteses testadas, bem como a escolhida, são mostradas na figura.

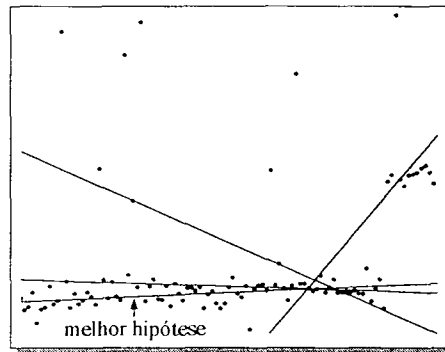


Figura 4.3: Hipóteses geradas pelo LMS para aproximação de reta

Um parâmetro crucial deste algoritmo é o número de subconjuntos, k , que deve ser grande o suficiente para proporcionar uma probabilidade alta de que ao menos um dos subconjuntos contenha somente *inliers* (pontos da superfície de interesse). Seja P_f a fração mínima de *inliers* esperada para o conjunto de pontos, e P_s a probabilidade desejada de que ao menos um dos subconjuntos s_j seja constituído apenas por *inliers*, o número mínimo de subconjuntos (hipóteses) a serem testados é dado pela equação 4.2

¹Como será visto na sequência, o valor da constante k pode sofrer grandes variações. Por isso o tempo de execução $O(kN)$ é enfatizado, ao invés de se considerar simplesmente $O(N)$.

[Fischler e Bolles, 1981].

$$k = \frac{\log(1 - P_s)}{\log(1 - P_f^n)} \quad (4.2)$$

A tabela 4.1 mostra os valores adequados para o número mínimo de hipóteses necessárias para se garantir uma boa probabilidade de acerto ($P_s = 0,99$) em aproximações de superfícies planas ($n = 3$), superfícies polinomiais biquadráticas ($n = 6$) e superfícies quádricas ($n = 9$). Como pode ser observado na tabela, o valor de k aumenta substancialmente conforme aumenta o número n , de pontos utilizados para gerar cada hipótese, e conforme diminui a porcentagem de *inliers*, P_f , no conjunto de dados.

n	P_f	k
3	0,5	35
3	0,25	293
6	0,5	293
6	0,25	18.861
9	0,5	2.356
9	0,25	1.207.216

Tabela 4.1: Número de hipóteses k para índice de confiança $P_s = 0,99$

Após a escolha da melhor hipótese, ainda é necessário realizar um refinamento dessa aproximação pois os parâmetros $\hat{\mathbf{a}}$ foram obtidos a partir de um subconjunto de apenas n pontos. Assim, é preciso utilizar a hipótese para determinar todos os *inliers* do conjunto de N pontos e realizar uma nova aproximação, possivelmente utilizando o LS e considerando apenas esses pontos *inliers*, para gerar o vetor de parâmetros final.

Considerando que o ruído presente na imagem segue uma distribuição normal de média $\mu = 0$ e desvio padrão σ , e obtendo uma estimativa $\hat{\sigma}$ para σ , então os pontos tais que $r_{i,\hat{\mathbf{a}}}^2 < (C\sigma)^2$, para uma constante C tipicamente em torno de 2,5, podem ser considerados como *inliers*. Este valor para C deve-se ao fato de que, em uma distribuição normal, 68% dos valores estão contidos no intervalo $[\mu - \sigma, \mu + \sigma]$, 95% em $[\mu - 2\sigma, \mu + 2\sigma]$ e 99,7% em $[\mu - 3\sigma, \mu + 3\sigma]$. A estimativa de *escala* denominada *Desvio Mediano Absoluto* (*Median Absolute Deviation*, MAD) pode ser obtida a partir da hipótese $\hat{\mathbf{a}}$ conforme mostra a equação 4.3,

$$\hat{\sigma} = 1 / \left(\Phi^{-1}(0.75) \left(1 + \frac{5}{N - n} \right) \sqrt{\text{median}_i \{r_{i,\hat{\mathbf{a}}}^2\}} \right) \quad (4.3)$$

onde Φ^{-1} é a inversa da distribuição normal cumulativa e $1 + 5/(N - n)$ é um fator de correção de amostra finita [Rousseeuw e Leroy, 1987].

Está última aproximação é muito importante pois utiliza todos os pontos, exceto os *outliers*, aumentando a eficiência estatística do LMS.

4.2.2 *Least Trimmed Squares*

O *Least Trimmed Squares* (LTS) [Rousseeuw e Leroy, 1987], ou “Mínimos Quadrados Cortados”, é um método semelhante ao LMS, porém mais eficiente. O LTS também possui *breakdown point* de 0,5 e utiliza amostragem aleatória, mas sua função objetivo é diferente – equação 4.4,

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmin}} \sum_{j=1}^h (r_{\mathbf{a}}^2)_{j:N} \quad (4.4)$$

onde $(r_{\mathbf{a}}^2)_{j:N}$, $1 \leq j \leq N$, são os quadrados dos resíduos da hipótese \mathbf{a} em ordem não-decrescente e, geralmente, $h = \lfloor (N+1)/2 \rfloor$ (o índice da mediana).

O LMS procura minimizar apenas a mediana dos quadrados dos resíduos e, desta forma, não é capaz de distinguir, corretamente, a melhor dentre várias hipóteses com valores de mediana muito próximos. Diferentemente, o LTS minimiza o somatório dos h menores resíduos – geralmente, a mediana e os valores menores que esta. Assim, um maior número de resíduos é levado em consideração e é esta a razão da maior eficiência.

Exceto pela função objetivo diferente, o funcionamento do LTS é semelhante ao do LMS e também podem ser realizadas a estimativa de escala ($\hat{\sigma}$) e classificação de *inliers* para a aproximação LS final.

4.2.3 *Random Sample Consensus*

O método *Random Sample Consensus* (RANSAC) [Fischler e Bolles, 1981], ou Consenso de Amostra Aleatória, é reconhecido, parcialmente, devido à sua semelhança ao método de estimação LMS, introduzido na área de estatística alguns anos mais tarde e adotado pela comunidade de visão computacional. O RANSAC também emprega uma técnica de amostragem aleatória e funciona da seguinte forma:

1. Dado um modelo que requer um número mínimo de n pontos para instanciar (determinar) seus parâmetros e um conjunto C contendo N pontos, com $N > n$, seleciona-se aleatoriamente um subconjunto S contendo n pontos e instância-se o modelo, gerando uma hipótese \mathbf{a} . A partir desta hipótese, determina-se o subconjunto S_c de pontos em C cujos resíduos são menores que um valor máximo de tolerância de erro, σ , previamente definido. Este conjunto S_c é denominado *conjunto de consenso* de S .
2. O processo acima é repetido várias vezes, selecionando-se aleatoriamente, a cada nova tentativa, um novo subconjunto S e armazenando-se S_c^* , o conjunto de consenso S_c que apresentou a maior cardinalidade, $|S_c|$, até o momento.
3. Após um número k de repetições, se $|S_c^*|$ for menor que um limiar t_c – que pode ser definido em função da quantidade relativa de *outliers* esperada para o conjunto C .

– considera-se que o processo falhou. Senão, os pontos em S_c^* são utilizados para estimar os parâmetros finais, possivelmente utilizando o método LS.

Cada conjunto de consenso S_c também pode ser considerado como o conjunto dos *inliers* de uma hipótese \mathbf{a} . Assim, a função objetivo do RANSAC pode ser definida como: encontrar a hipótese que maximize o número de *inliers* – os pontos apresentando valor residual menor que σ – ou, equivalentemente, que minimize o número de *outliers*. Está última é a definição normalmente considerada para a função objetivo do RANSAC, mostrada na equação 4.5,

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmin}} \sum_i \rho(r_{i,\mathbf{a}}) \quad (4.5)$$

onde o termo de erro $\rho()$ é definido como

$$\rho(r) = \begin{cases} 0 & \text{se } |r| \leq \sigma \\ 1 & \text{se } |r| > \sigma \end{cases} \quad (4.6)$$

A idéia de maximizar o número de pontos que concordam com a hipótese de superfície justifica-se pelo fato de o quanto maior for esse número, menor a possibilidade de tal configuração de pontos ser aleatória, e maior a chance da hipótese ser válida.

Os três parâmetros que devem ser previamente especificados para o método são: (1) a tolerância de erro usada para determinar se um ponto é ou não compatível com o modelo, ou seja, a largura da faixa de *inliers* (*inlier bound*), como mostrado na figura 4.4; (2) o número k de tentativas ou de hipóteses geradas (calculado da mesma forma que para o LMS); (3) o limiar t_c , a cardinalidade no menor conjunto de *inliers* aceitável.

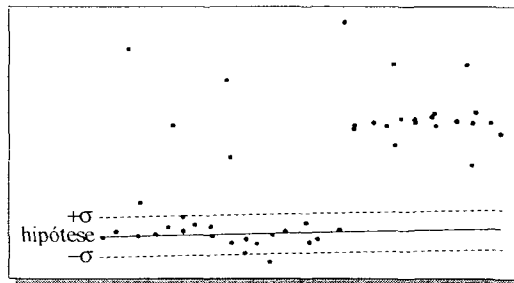


Figura 4.4: Tolerância de erro (faixa de inliers) do RANSAC

O parâmetro σ do RANSAC tem a mesma função que a escala $\hat{\sigma}$, estimada para o LMS e para o LTS (equação 4.3). A diferença é que no RANSAC este é um parâmetro pré-especificado, enquanto nos outros dois métodos este parâmetro é estimado a partir de valores residuais. Ou seja, os métodos LMS e LTS são capazes de gerar as hipóteses sem requerer que esse parâmetro esteja disponível inicialmente. Somente na identificação dos *inliers*, para a aproximação final, é que o valor estimado para a escala é necessário. Já no RANSAC, todas as hipóteses são geradas utilizando o limite de tolerância previamente especificado.

Muitas vezes, esse limite de tolerância de erro do RANSAC pode ser estimado experimentalmente para um conjunto de imagens providas por um mesmo sensor. Isso é realizado aproximando-se modelos aos dados das imagens, calculando-se os valores residuais, seu desvio-padrão ($\hat{\sigma}$) e, então, ajustando-se a tolerância de erro, σ , de modo que $\sigma \approx 2,5\hat{\sigma}$.

Interessantemente, por virtude dessa faixa de *inliers* pré-especificada, o RANSAC pode ser utilizado para encontrar estruturas formadas por um número de pontos substancialmente menor que 50% do conjunto de dados C . Ou seja, o método possui um valor de *breakdown point* maior que 0,5.

4.2.4 *M-estimator Sample Consensus*

Um problema apresentado pelo método RANSAC é que, se ao limiar σ for atribuído um valor muito alto, a eficiência do método pode ser bastante afetada já que este não é capaz de identificar a melhor hipótese dentre diferentes hipóteses com o mesmo número de *outliers*.

Na função objetivo do RANSAC, mostrada na equação 4.6, os *inliers* representam um custo de erro nulo, enquanto que os *outliers* representam um custo de erro igual a uma constante, neste caso o valor 1. Quanto maior for o valor σ , mais hipóteses apresentarão um mesmo valor para o somatório dessa equação, produzindo uma estimativa incorreta. Por exemplo, se σ for suficientemente grande, então todas as hipóteses terão o mesmo custo (zero) pois todos os pontos podem ser considerados como *inliers*.

Este problema pode ser evitado sem a introdução de nenhum custo computacional adicional, alterando-se o termo de erro $\rho()$ da equação 4.6 para:

$$\rho(r) = \begin{cases} r & \text{se } |r| \leq \sigma \\ \sigma & \text{se } |r| > \sigma \end{cases} \quad (4.7)$$

Este novo termo é semelhante a um operador do tipo *M-estimator* [Press *et al.*, 1992] e, por isso, esta versão alterada do RANSAC é chamada de *M-estimator Sample Consensus*, o método MSAC [Torr e Zisserman, 2000]. Este método objetiva a minimização do número de *outliers*, enquanto minimiza o somatório dos resíduos dos *inliers*. Ou seja, para os *outliers* continua sendo atribuído um valor de penalidade fixo (σ), mas os *inliers* são agora considerados de acordo com o quão bem eles se ajustam ao modelo. O MSAC aumenta a eficiência do RANSAC adicionando absolutamente nenhum fardo computacional extra. Por isso, seus autores defendem que não existe nenhum motivo para usar o RANSAC ao invés do MSAC.

4.2.5 Discussão

O RANSAC e o LMS são dois métodos de estimação robustos muito significativos que foram introduzidos na área de visão computacional e inspiraram vários outros métodos desenvolvidos nesta área de pesquisa.

O RANSAC utiliza um valor de escala previamente definido, enquanto o LMS estima esse valor, durante a aproximação, a partir do conjunto de dados. Em aplicações de visão computacional, esta diferença fundamental é muitas vezes esquecida por causa da natureza praticamente idêntica das técnicas de amostragem aleatória utilizadas pelos dois métodos durante o processo de minimização.

Atualmente, métodos derivados do RANSAC tem substituído com freqüência o LMS nos algoritmos de visão pois o fato de ter dois parâmetros de ajuste adicionais (a cardinalidade mínima do conjunto de *inliers* para a aceitação dos parâmetros estimados e a quantidade de ruído permitida para os pontos desse conjunto), além de um valor de *breakdown point* significativamente maior que 0,5, torna-o de mais fácil adaptação a algumas situações específicas de análise de dados.

Apesar de o LMS apresentar *breakdown point* menor, de 0,5, alguns métodos derivados do LMS [Lee *et al.*, 1998] [Stewart, 1995] [Miller e Stewart, 1996] podem apresentar resultados tão bons quanto, e até melhores que, os do RANSAC sem requerer a pré-especificação de um valor de faixa de *inliers*.

No entanto, ainda existem casos em que, mesmo utilizando métodos robustos, a aproximação pode ser corrompida. Por exemplo, a figura 4.5 mostra o perfil (corte transversal) de uma região de borda em forma de degrau entre duas superfícies planas, onde a aproximação de uma das superfícies foi corrompida por pontos da outra superfície (*pseudo-outliers*). Experimentos realizados mostram que, atualmente, para os métodos robustos mais eficientes, isso ainda pode ocorrer se a altura do degrau for menor que $4,5\sigma$, onde σ é o desvio-padrão do ruído presente nos dados. A maioria das técnicas de estimativa robustas mais simples, incluindo o LMS, falham quando a magnitude desse tipo de descontinuidade é menor que $7,4\sigma$ [Stewart, 1999]. Portanto, ainda há a necessidade de aprimoramento dos métodos existentes ou mesmo de desenvolvimento de novos métodos robustos no campo de visão computacional.

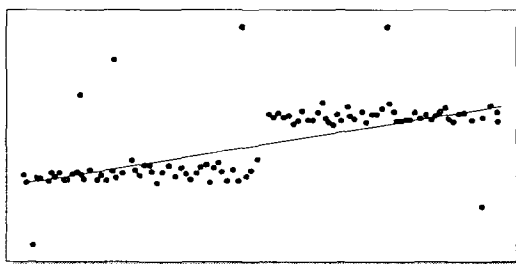


Figura 4.5: Aproximação corrompida em descontinuidade de pequena magnitude

4.3 Extração de Primitivas Geométricas

A extração de primitivas geométricas como curvas, círculos e superfícies é um problema de grande importância no contexto da visão computacional baseada em modelos. Como os contornos e as superfícies dos objetos podem ser descritos como primitivas geométricas, a extração destas primitivas é, portanto, um passo considerável rumo à identificação e localização de objetos no ambiente observado.

O problema de extração de primitivas geométricas tem sido tratado do ponto de vista da área de estimativa robusta. Métodos de aproximação tradicionais, como o LS, não são capazes de extrair primitivas porque consideram que todos os pontos do conjunto de dados descrevem uma única primitiva. Entretanto, o uso de métodos de aproximação robustos possibilita a identificação de um subconjunto de pontos, os *inliers*, e a associação deste subconjunto a uma única primitiva. Estes *inliers* podem ser desconsiderados nos passos seguintes, quando as demais primitivas restantes na imagem estiverem sendo procuradas. Assim, métodos de estimação robustos podem ser utilizados não apenas para aproximação, mas também para auxiliar na solução do problema mais difícil que é a extração.

Alguns métodos de extração como a Transformada Hough [Jain *et al.*, 1995] podem extrair primitivas representadas por uma porcentagem muito pequena do conjunto de dados e de uma forma muito eficiente em termos de tempo de computação. No entanto, essa técnica é muito ineficiente quanto à utilização de memória, já que os requisitos estão em função exponencial em termos da dimensão do espaço de parâmetros da primitiva. Por isso, a maioria das aplicações que se utilizam da Transformada Hough empregam-na para a extração de primitivas com no máximo dois graus de liberdade (parâmetros), como por exemplo para a extração de linhas em imagens bidimensionais.

Já os métodos de estimativa robusta, discutidos na seção anterior, apresentam baixos requisitos de espaço e são lineares em relação ao número de pontos, ao invés de exponenciais em relação ao número de parâmetros, o que é uma grande vantagem em relação à Transformada Hough. Isso possibilita a extração de primitivas com vários graus de liberdade (e.g. superfícies complexas).

Roth e Levine demonstram em [Roth e Levine, 1993] que a extração de primitivas é essencialmente um problema de otimização no qual o objetivo é encontrar o mínimo global de uma função de custo robusta, a qual potencialmente possui muitos mínimos locais. Para uma extração eficiente, o verdadeiro mínimo global deve ser encontrado com o menor número possível de avaliações da função de custo. Segundo este modelo, é possível extrair confiavelmente primitivas descritas por menos que 50% dos dados da imagem. A abordagem proposta em [Roth e Levine, 1993] emprega a técnica de amostragem aleatória de conjuntos mínimos e o resultado da extração é a primitiva apresentando o menor valor para a função de custo.

Um conjunto mínimo é o menor conjunto de pontos necessários para definir uma única instância de uma primitiva geométrica. O primeiro algoritmo a introduzir a utilização de

conjuntos mínimos para a extração de primitivas geométricas foi o RANSAC, descrito na seção 4.2.3. Para dados com baixo nível de ruídos, um conjunto mínimo freqüentemente produz uma boa representação para todos os pontos da primitiva. Outro ponto importante é que somente conjuntos de parâmetros definidos por conjuntos mínimos são possíveis soluções para o processo de extração. Se conjuntos mínimos não fossem utilizados, então cada combinação de valores possível para o conjunto de parâmetros teria de ser considerada. Portanto, a utilização de conjuntos mínimos reduz muito o espaço de busca do algoritmo de extração.

Como existem muitas combinações de pontos formando conjuntos mínimos, avaliar a todos eles garante a obtenção do mínimo global, contudo não é prático em termos de custo computacional. A utilização da amostragem aleatória torna possível avaliar um número muito menor de conjuntos e ainda assim obter sucesso.

No entanto, como mostra a tabela 4.1, o número de hipóteses necessárias (k) para a obtenção de um resultado confiável pode aumentar rapidamente conforme diminui a porcentagem de *inliers* (e.g. o conjunto de pontos da imagem descreve um maior número de superfícies) e aumenta o número de pontos necessários para instanciar o modelo da primitiva geométrica. Deste modo, o tempo de execução dominante do algoritmo consiste na avaliação da função de custo para cada uma das k hipóteses, o que é realizado em tempo $O(kN)$, onde N é o número total de pontos no conjunto de dados. Como normalmente não se pode otimizar ainda mais a função de custo, uma solução para diminuir o custo computacional consiste em diminuir o número de hipóteses analisadas, porém, mantendo o grau de confiabilidade.

Uma abordagem atraente é primeiramente dividir os dados em regiões conexas – possivelmente utilizando informações de bordas, extraídas a um custo computacional relativamente baixo – e aplicar o processo de amostragem aleatória a cada região em separado. Deste modo, serão calculados os valores residuais para um número menor de pontos. Além disso, como cada região conterá um número menor de primitivas, menor poderá ser percentual de *outliers* dentro dessa região e, portanto, poderá ser menor também o número de hipóteses necessárias para que a extração das primitivas nessa região seja bem sucedida.

O fato de o processo avaliação das hipóteses ser facilmente paralelizável também contribui para diminuir o tempo de resposta de métodos de extração de primitivas. Ainda, a técnica aleatória também pode ser substituída por algoritmos genéticos, mais eficientes e descritos a seguir.

4.3.1 Algoritmos Genéticos

Em muitos problemas de otimização, o número de ótimos locais pode ser muito grande e encontrar o ótimo global pode ser uma tarefa de grande custo.

Um algoritmo genético [Man *et al.*, 1996] é uma abordagem de otimização baseada

na metáfora da evolução das espécies. Tais algoritmos apresentam uma performance comprovadamente melhor que técnicas de busca aleatória para a solução de problemas de otimização *hard*. Um problema de otimização é *hard* se a função de custo é ruidosa, multi-dimensional, e possui muitos ótimos locais. Como a extração de primitivas geométricas é considerada um problema desta natureza, o uso de um algoritmo genético mostra-se apropriado.

4.3.1.1 Descrição

Os conceitos relacionados a algoritmos genéticos são considerados simples e sua compreensão não requer formalização matemática como no caso de outros métodos de otimização.

Algoritmos genéticos (AGs) são inspirados no mecanismo de seleção natural, um processo biológico no qual, em um ambiente de competição, indivíduos mais adaptados possuem maiores chances de sobrevivência. Assim, AGs empregam uma analogia direta da teoria da evolução natural de Darwin.

Considera-se um número de possíveis soluções para o problema como *indivíduos* de uma *população*. Cada indivíduo é representado por um conjunto de parâmetros, os *genes* de um *cromossomo*, o qual geralmente é estruturado em uma lista seqüencial (*string*) de valores. A cada indivíduo é associado um valor positivo, chamado de valor de adaptação (*fitness*), o qual é utilizado para avaliar a qualidade da solução representada pelo cromossomo. Este valor está relacionado com o valor da função objetivo definida para o problema. Os passos de execução descritos abaixo são mostrados no diagrama da figura 4.6.

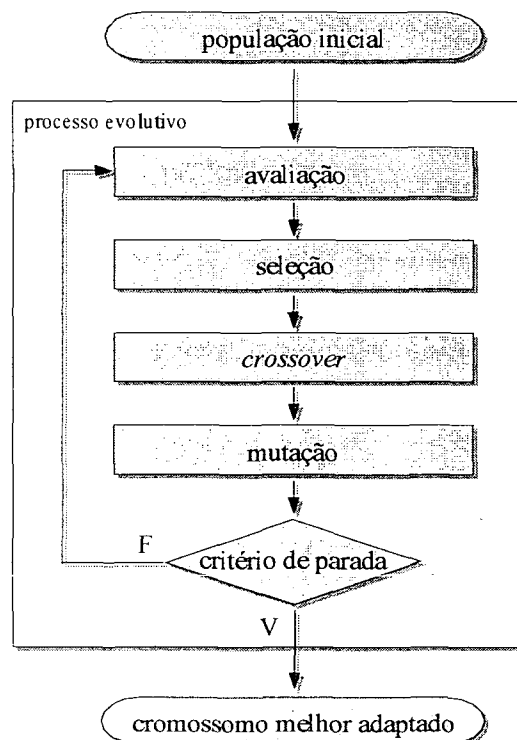


Figura 4.6: Diagrama de execução de um algoritmo genético.

Primeiramente, uma população inicial de cromossomos é criada (e.g. de forma aleatória). Seu tamanho, normalmente fixo durante o processo de evolução, é estabelecido de acordo com a natureza do problema a ser resolvido. A cada ciclo do processo evolutivo, uma nova geração (população) de indivíduos é criada a partir dos cromossomos da geração anterior. Este processo evolucionário é repetido até que uma condição de parada seja alcançada, como, por exemplo, um número máximo de gerações ou indicativo de convergência para o ótimo global (e.g. mudanças não significativas no valor de adaptação do melhor indivíduo durante as últimas n gerações). Quando a condição de parada é obtida, o indivíduo apresentando a melhor adaptação é considerado como a solução encontrada para o problema.

Cada nova geração é obtida, da geração anterior, através da aplicação de quatro operações genéticas:

1. A operação de *avaliação* ter por objetivo medir o valor de adaptação de cada indivíduo, de acordo com a qualidade da solução do problema por ele representada.
2. A operação de *seleção* escolhe casais de cromossomos, os pais, para que possa haver a reprodução. Nesta operação, quanto maior o valor de adaptação de um cromossomo maiores são as suas chances de ser escolhido. Assim, os melhores cromossomos geram um maior número de descendentes e seus genes tem maiores chances de sobrevivência na geração seguinte.
3. *crossover*, ou troca genética, é uma operação também aplicada a pares de cromossomos e ilustrada na figura 4.7(a). Nesta operação, partes de cromossomos “pais”, definidas (aleatoriamente ou deterministicamente) utilizando pontos de *crossover*, são recombinadas para gerar dois cromossomos “filhos”. Cada parte pode ser trocada com uma probabilidade de *crossover*, P_c , tipicamente entre 0,6–1,0. A justificativa para a realização desta operação é que espera-se que a combinação de dois indivíduos bem adaptados possa gerar ao menos um filho cuja adaptação seja ainda melhor. Ou seja, a operação de *crossover* utiliza soluções parciais como blocos de construção que são combinados para criar soluções melhores.
4. A *mutação* é uma operação aplicada, individualmente, a cada cromossomo filho. Como mostra a figura 4.7(b), esta operação pode alterar, aleatoriamente, cada um dos genes com uma probabilidade pequena, P_m , normalmente menor que 0,1. A finalidade desta operação é introduzir novo material genético na população, mas com uma pequena probabilidade para evitar que a evolução degenere para uma busca aleatória.

A escolha de valores adequados para as probabilidades P_c e P_m é criticamente dependente da natureza da função objetivo e ainda permanece como uma questão em aberto, apesar de algumas sugestões serem apresentadas na literatura [Man *et al.*, 1996].

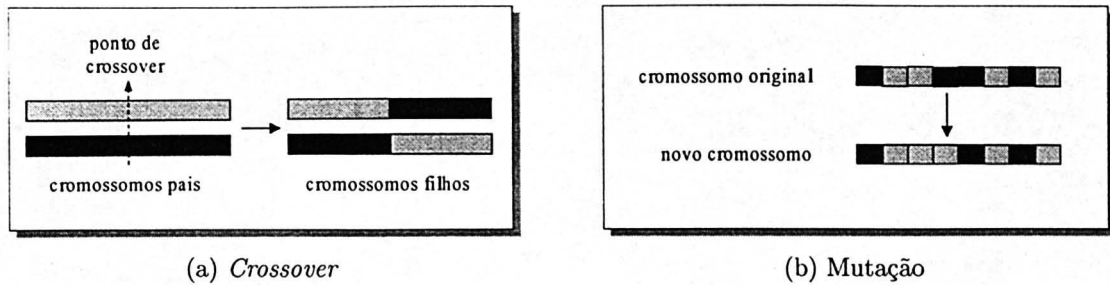


Figura 4.7: Operações genéticas de crossover e mutação.

Existem também cuidados que se deve tomar quanto à implementação da operação de seleção *seleção*. Caso ela priorize demais os indivíduos mais adaptados, a população pode se transformar rapidamente numa “elite” de indivíduos com baixa variedade genética, causando uma convergência precoce para um ótimo local. Um estudo comparativo de métodos de seleção é encontrado em [Zhang e Kim, 2000].

4.3.1.2 Aplicação na Extração de Primitivas Geométricas

Métodos de extração de primitivas [Roth e Levine, 1994] [Yu *et al.*, 1994] [Chen e Liu, 1999] empregam AGs com o objetivo de acelerar o processo de busca pela melhor hipótese, ou seja, de modo extrair uma primitiva com o mesmo grau de confiança, mas analisando um menor número de hipóteses que o requerido na busca aleatória. Além disso, AGs também apresentam um alto grau de paralelismo. Estes dois fatores conjugados são de extrema importância para a viabilização da extração de primitivas com vários graus de liberdade.

A representação de cromossomos utilizada pelos algoritmos de extração de primitivas consiste em um vetor de genes, cada um contendo um valor inteiro referente ao índice de um *pixel* da imagem. Seja o par (i, j) as coordenadas de linha e coluna de um *pixel* qualquer e L a largura da imagem, o índice associado a este *pixel* é dado por: $I(i, j) = i \times L + j$. Assim, na extração de planos, por exemplo, cada cromossomo contém três valores inteiros (três genes) correspondentes a três *pixels* da imagem.

A função objetivo empregada deve primeiro instanciar uma hipótese a partir dos pontos contidos no cromossomo e em seguida fornecer um valor de custo para a primitiva descrita por esses pontos. Como valores de custo devem ser minimizados e, por outro lado, AGs procuram maximizar valores de adaptação (*fitness*), faz-se necessário realizar um mapeamento entre esses dois tipos de valores, custo e adaptação. Uma fórmula geralmente utilizada para isso é: $fitness = \frac{1}{custo}$. Desta forma, ao maximizar-se o valor de adaptação, minimiza-se o valor de custo.

Métodos de extração de primitivas empregando busca aleatória ou AGs para fins de segmentação de imagens são descritos em maiores detalhes na próxima seção.

4.4 Trabalhos Relacionados

Esta seção discute alguns dos métodos mais relevantes, encontrados na literatura, para a segmentação de imagens baseada na extração de superfícies. Os métodos apresentados estão relacionados em duas categorias, de acordo com o tipo de otimização utilizado: amostragem aleatória ou algoritmos genéticos.

4.4.1 Otimização por Amostragem Aleatória

Os métodos de extração de primitivas discutidos a seguir [Roth e Levine, 1993] [Lee *et al.*, 1998] utilizam a técnica de amostragem aleatória na definição de subconjuntos de pontos utilizados para instanciar o modelo (hipóteses) das superfícies a serem extraídas.

4.4.1.1 Roth e Levine

Roth e Levine apresentam em [Roth e Levine, 1993] um modelo de otimização para a extração de primitivas geométricas em imagens como uma generalização do método RANSAC (seção 4.2.3). Segundo este modelo, baseado em conjuntos mínimos e na técnica de amostragem aleatória, a extração de uma primitiva consiste na minimização de uma função de custo. Para uma extração eficiente, o mínimo global deve ser encontrado com o menor número possível de avaliações da função de custo.

O algoritmo apresentado pode ser utilizado para extrair múltiplas primitivas de um mesmo tipo, ou primitivas de tipos diferentes, simultaneamente. O trabalho discute ainda assuntos importantes como: escolha do número de amostras aleatórias necessárias na otimização; métodos para instanciar o modelo de superfície a partir de conjuntos mínimos; e também formas para acelerar a avaliação da função de custo utilizando processamento paralelo. Entre os experimentos apresentados, é analisada em maiores detalhes, a seguir, a extração de superfícies planas e quádricas a partir de imagens de profundidades.

O algoritmo utiliza como modelos de superfície as formas implícitas de planos e quádricas, mostradas nas equações 4.8 e 4.9.

$$f(x, y, z) = ax + by + cz + d = 0 \quad (4.8)$$

$$f(x, y, z) = a_1x^2 + a_2y^2 + a_3z^2 + a_4xy + a_5xz + a_6yz + a_7x + a_8y + a_9z + a_{10} = 0 \quad (4.9)$$

Para determinar os n parâmetros de um modelo, o algoritmo utiliza um conjunto mínimo contendo $n - 1$ *pixels* da imagem de profundidade, cada qual associado a um ponto 3D de coordenadas (x_i, y_i, z_i) . Amostras aleatórias de três pontos são utilizados para determinar hipóteses de planos. Amostras de nove pontos são utilizadas para hipóteses de quádricas. Substituindo x , y e z , na forma geral da superfície, pelas coordenadas (x_i, y_i, z_i) de cada ponto e considerando as equações resultantes juntamente com a forma geral, é obtido um sistema linear de n equações da forma $\mathbf{M}\mathbf{a} = \mathbf{0}$, onde \mathbf{M} é uma

matriz quadrada, \mathbf{a} é um vetor coluna com os parâmetros a serem determinados, e $\mathbf{0}$ é o vetor coluna nulo. Como uma solução não trivial para esse sistema existe somente se o determinante da matriz \mathbf{M} é igual a zero, os elementos do vetor de parâmetros \mathbf{a} são determinados expandindo o determinante da matrix, reunindo os termos da forma geral e igualando a zero.

A função de custo do RANSAC é utilizada então para contar o número de pontos que estão dentro da faixa de *inliers* determinada pelo parâmetro σ , o qual é definido como o valor estimado para o desvio padrão do ruído nos dados. Para contar o número de *inliers*, é necessário calcular a distância geométrica (Euclideana) entre cada ponto e a superfície determinada pelo vetor de parâmetros obtido. No entanto, para superfícies mais complexas, como quádricas, não existe fórmula fechada para o cálculo de tal distância. De modo a evitar, nesse cálculo, a utilização de métodos iterativos, o algoritmo emprega uma aproximação de primeira ordem [Taubin, 1991] para a distância geométrica. Esta aproximação é obtida da razão entre o valor absoluto da função e a magnitude do vetor gradiente no ponto (x_i, y_i, z_i) em questão (equação 4.10).

$$dist \approx \frac{|f(x_i, y_i, z_i; \mathbf{a})|}{|\nabla f(x_i, y_i, z_i; \mathbf{a})|} \quad (4.10)$$

Para diminuir a quantidade de dados a serem processados na extração de cada primitiva, a imagem primeiramente é decomposta em regiões conexas definidas por bordas de profundidade e orientação. A maior região conexa é passada então ao algoritmo que extrai as superfícies planas e quádricas contidas na região. Para cada região, o algoritmo tenta extrair os dois tipos de superfícies. As hipóteses para o melhor plano e a melhor quádrica são analisadas e aquela que apresentar o maior número de *inliers* é aceita. Estes *inliers* são removidos da região e o processamento continua enquanto o tamanho da região for maior que um tamanho mínimo, previamente estipulado. Quando não há pontos suficientes nessa região, ela é descartada e o processo é reiniciado para a maior região conexa entre as restantes.

Este foi o primeiro trabalho a generalizar um esquema para a extração de primitivas 2D e 3D, e a discutir as questões envolvidas nesse processo. As seguintes observação são feitas em relação ao método proposto por Roth e Levine:

1. A detecção de bordas de orientação, realizada para reduzir o espaço de busca das extrações, apesar de reduzir consideravelmente o número de hipóteses necessárias, é um processo que requer uma aproximação de superfície local (geralmente LS) para a estimativa de vetores normais, o que é uma tarefa sensível a *outliers*. Assim, as bordas detectadas podem não preservar as formas dos objetos e o mesmo acontecerá com as superfícies extraídas. O trabalho também não discute a questão do fechamento do mapa de bordas obtido (nenhum método de fechamento é utilizado), de modo que se houver um intervalo (lacuna) de apenas um ponto no mapa de bordas, regiões distintas podem permanecer incorretamente conexas,

diminuindo os benefícios da detecção de bordas inicial.

2. A aproximação utilizada para a distância entre pontos e a hipótese da superfície, além de não corresponder à verdadeira distância geométrica, apresenta instabilidade em pontos onde o gradiente é muito próximo de zero. Isto faz com que o valor resultante da equação 4.10 seja incorretamente grande e impreciso.
3. Métodos que, como este, procuram apenas maximizar o número de pontos dentro da faixa de *inliers*, sem levar em consideração o erro destes pontos, podem gerar hipóteses finais imprecisas pois a faixa de *inliers* tende a ser deslocada o máximo possível para que contenha um maior número de pontos, conforme mostrado na figura 4.8(a). Assim, além de a melhor hipótese não corresponder à aproximação desejada, um maior número de pontos de outras superfícies são incorretamente classificados como *inliers* da superfície sendo extraída. A figura 4.8(b) mostra que este problema pode ser atenuado com a substituição da função de custo do RANSAC pela função de custo do MSAC (seção 4.2.4, página 70), que leva em consideração as distâncias dos *inliers* em relação à hipótese de superfície.
4. A extração de superfícies quádricas utilizando a técnica de amostragem aleatória apresenta um custo computacional muito alto. Como as hipóteses de superfícies devem ser instanciadas a partir de nove pontos, um número muito grande de avaliações da função de custo é necessário para garantir que ao menos umas das hipóteses geradas seja constituída apenas por pontos “bons”. Por exemplo, considerando um grau de confiabilidade de 95% e uma expectativa mínima de 40% de *inliers* (valores bastante otimistas), seria necessário avaliar 11427 hipóteses, segundo a equação 4.2 (página 67), para cada superfície extraída.

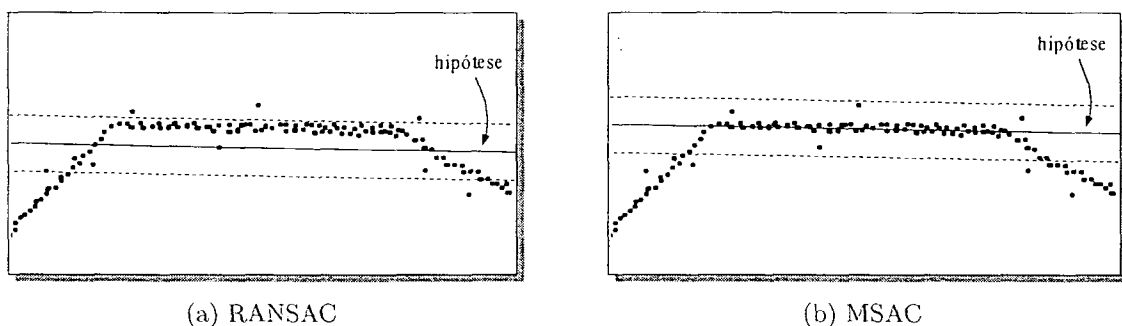


Figura 4.8: Diferenças no comportamento das faixas de *inliers* do RANSAC e do MSAC.

4.4.1.2 Lee, Meer e Park

Em [Lee *et al.*, 1998], Lee *et al.* propõem o método de estimativa robusto *Adaptive Least Kth order Squares* (ALKS), derivado do LMS (seção 4.2.1, página 65), além de

um algoritmo para a segmentação de imagem baseado neste método e na técnica de amostragem aleatória.

Segundo os autores, quando a segmentação é baseada apenas em um modelo de superfície plana, superfícies com vários graus de liberdade podem ser satisfatoriamente aproximadas caso sejam utilizados estimadores robustos com alto índice de *breakdown*. Se necessário, as regiões planas identificadas podem posteriormente ser unidas utilizando modelos de superfície de maior ordem. Ainda, um modelo com maior número de graus de liberdade frequentemente pode gerar subsegmentação, ou seja, a união de duas regiões adjacentes. Por exemplo, duas superfícies separadas por uma borda de orientação podem ser unidas na utilização de um modelo de superfície quadrático. Além disso, o estudo sistemático mais significativo de comparação de métodos de segmentação [Hoover *et al.*, 1996] foi aplicado apenas a segmentadores de superfícies planas.

O algoritmo de segmentação de imagens apresentado pode ser resumido como segue abaixo:

1. Defina a região a ser processada como a maior região conexa de pontos ainda não rotulados. Inicialmente, isso corresponde à imagem toda.
2. Aplique o método ALKS à região selecionada para a identificação de *inliers*.
3. Atribua um rótulo à (extraia a) maior região conexa de *inliers* e considere-a como um esboço da região extraída.
4. Realize um refinamento dos parâmetros estimados para o modelo utilizando uma aproximação LS aos *inliers* da região esboço.
5. Repita os passos 1–5 até que o tamanho da maior região conexa identificada na iteração seja menor que um valor de limiar (100 pontos nos experimentos apresentados).
6. Elimine os pontos não rotulados restantes na imagem. Estes pontos são atribuídos a regiões vizinhas, determinadas pela maioria dos rótulos dos pontos vizinhos.

Um ponto positivo deste algoritmo de segmentação é que os únicos parâmetros necessários são o tamanho mínimo de região e o número de amostras aleatórias.

Nos resultados experimentais apresentados, o processo de segmentação utilizou 500 amostras aleatórias por região plana segmentada. Bordas um tanto distorcidas foram formadas na junção das superfícies, o que pode dificultar na identificação das formas dos objetos. O algoritmo também falhou em detectar corretamente algumas regiões, que acabaram sendo unidas, de modo que a segmentação produzida ainda requer uma outra etapa de processamento para gerar a segmentação final. Os autores propoem reprocessar regiões apresentando erros de aproximação grandes, aplicando novamente o método de extração, desta vez com maior número de hipóteses aleatórias compreendendo pontos apenas das regiões unidas.

Nota-se que em imagens com alto nível de ruídos, os *inliers* identificados no passo 2 podem formar uma nuvem esparsa, desconexa, com várias regiões pequenas. Desta forma, o passo 3 deveria ser redefinido para evitar a quebra de uma mesma região plana em várias regiões menores. Assim, não seria causada a segmentação incorreta de superfícies da mesma ordem que o modelo utilizado.

4.4.2 Otimização por Algoritmo Genético

A seguir são apresentados os métodos de extração de superfícies [Roth e Levine, 1994] [Chen e Liu, 1999] [Yu *et al.*, 1994] utilizando algoritmos genéticos no processo de otimização de hipóteses.

4.4.2.1 Roth e Levine

Na extração de superfícies, como visto anteriormente (tabela 4.1, página 67), caso existam muitas superfícies em uma região da imagem (i.e. caso a porcentagem *inliers* para cada superfície seja pequena), ou caso o modelo de superfície utilizado tenha uma maior número de graus de liberdade, então um grande número de amostras aleatórias será necessário. De modo a abordar especificamente este assunto, Roth e Levine propõem em [Roth e Levine, 1994] um algoritmo de extração baseado em um algoritmo genético (AG).

Este algoritmo é um aperfeiçoamento do algoritmo apresentado em [Roth e Levine, 1993] (seção 4.4.1.1) e as mudanças introduzidas afetam apenas o processo de determinação da melhor hipótese dos modelos de superfície (plano e quádrlica) para uma determinada região da imagem. No algoritmo descrito a seguir, a técnica de amostragem aleatória é substituída por um AG, sendo que os estágios restantes da segmentação permanecem os mesmos.

No AG utilizado, cada conjunto mínimo é representado por um cromossomo, no qual cada gene contém um índice inteiro que representa um ponto da imagem. Este foi o primeiro algoritmo de extração a empregar um AG no qual cromossomos representam conjuntos mínimos de pontos.

A função de custo recebe como parâmetro de entrada um cromossomo e fornece como saída um valor escalar correspondente ao valor de adaptação deste indivíduo. Quanto maior este valor, mais adaptado é o indivíduo. Assim, o valor utilizado como valor de *fitness* é o número de *inliers*, obtido com a função de custo do RANSAC, para a hipótese de superfície determinada pelos pontos nos genes do cromossomo.

No processo de extração das primitivas (superfícies) em uma determinada região da imagem, a população de cromossomos inicial é criada por amostragem aleatória. Então, as iterações de reprodução são iniciadas e somente terminam quando é alcançado um número máximo de iterações consecutivas sem que tenha surgido um novo indivíduo cujo valor de adaptação seja o melhor entre todos na população. Ou seja, o processo termina.

ao ser detectado um indicativo de convergência. A cada iteração, apenas um par de cromossomos é selecionado para a reprodução e então a operação de *crossover* é aplicada a estes dois indivíduos. Os dois filhos resultantes são avaliados e inseridos na população. Para manter o tamanho da população fixo, os dois indivíduos com os piores valores de adaptação são descartados.

O operação de seleção utilizada atribui probabilidades de escolha proporcionais aos valores de adaptação dos indivíduos de modo que os mais adaptados tem maiores chances de reproduzir.

O algoritmo emprega um operador de *crossover uniforme*, o qual considera cada gene (ponto da imagem) como uma possível parte a ser trocada, de acordo com a probabilidade de *crossover* P_c . Ou seja, no *crossover* uniforme, existe um ponto de *crossover* entre cada par de genes vizinhos. Esta operação é ilustrada na figura 4.9.

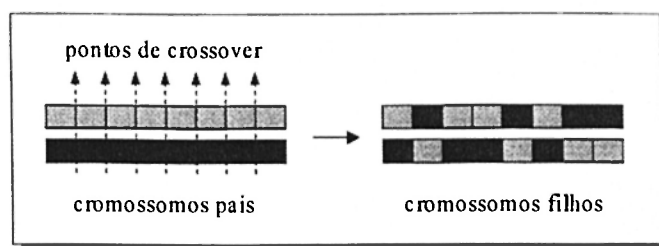


Figura 4.9: Operação de crossover uniforme.

Os resultados experimentais apresentados mostram uma grande melhoria em relação à versão do algoritmo utilizando otimização aleatória. O emprego do AG permite a obtenção de melhores resultados para a função de adaptação requerendo um número consideravelmente menor de avaliações da função de custo. A diferença entre as duas abordagens é consideravelmente maior quando existem mais primitivas de superfícies na região a ser processada e quando as primitivas possuem um maior número de graus de liberdade. Ou seja, quanto mais complexa a cena, melhor é o desempenho do AG em relação à amostragem aleatória.

A operação de crossover é a principal responsável por essa maior eficiência. Na amostragem aleatória, é preciso considerar um número de hipóteses, geralmente grande, para garantir que ao menos um dos conjuntos mínimos gerados será composto por pontos “bons” e de uma mesma superfície. Já nos AGs, o operador de *crossover* aliado à maior probabilidade de seleção dos indivíduos mais adaptados, permite combinar os pontos “bons” de diferentes cromossomos acelerando o processo de busca.

Outra vantagem é que, em vez de utilizar a abordagem geracional na qual a cada iteração uma nova população é criada e a anterior é descartada, o algoritmo emprega uma abordagem chamada *steady-state* (“estado estacionário”), na qual a cada iteração ocorre apenas uma reprodução e, do total de indivíduos, apenas os dois menos adaptados são descartados. Esta abordagem apresenta maior rapidez de convergência porque os filhos gerados em uma reprodução são imediatamente inseridos na população atual com

possibilidade de reproduzir na próxima iteração. Na abordagem geracional, os filhos gerados só poderam reproduzir depois que toda a nova população estiver formada. Além disso, pode ocorrer de o melhor indivíduo da nova população apresentar um valor de adaptação pior que o melhor indivíduo da população anterior. Na abordagem *steady-state* o melhor indivíduo sempre é mantido na população.

Como pontos negativos do AG utilizado pode-se citar os seguintes fatores que levam à perda de variedade genética e que aumentam a possibilidade de ocorrer convergência prematura:

1. A utilização de seleção proporcional, na qual a população tende a ser dominada por uma elite de indivíduos. Muitos trabalhos [Man *et al.*, 1996] [Zhang e Kim, 2000] [Whitley, 1989] demonstram que melhores resultados podem ser obtidos com a utilização de operações de seleção baseadas em ranking ou torneio [Zhang e Kim, 2000].
2. É permitida a existência de indivíduos idênticos na população.
3. Não é utilizado um operador de mutação, o que poderia introduzir novo material genético.

As observações feitas para o método da seção 4.4.1.1, o qual utiliza amostragem aleatória, também se aplicam a este, com exceção da observação número 4 pois o AG utilizado faz com que o número de avaliações de hipóteses necessárias seja consideravelmente reduzido.

4.4.2.2 Chen e Liu

O objetivo do método de extração de superfícies proposto por Chen e Liu em [Chen e Liu, 1999] é a reconstrução de modelos CAD (*Computer-Aided Design*) – aplicação denominada *engenharia reversa* – de objetos 3D compostos por superfícies planas e quádricas.

Uma única representação de superfície quádrica (equação 4.9) é utilizada pelo método. A extração de superfícies é realizada empregando um AG e cada superfície extraída é classificada em um tipo específico de superfície quádrica (e.g. plano, cone, cilindro) de acordo com uma tabela de classificação baseada nos parâmetros estimados para a superfície.

Um método rápido para aproximação LS de superfícies quádricas, baseado em multiplicadores de Lagrange e decomposição de matrizes em autovalores, é utilizado para instanciar hipóteses de quádricas a partir dos nove pontos em cada cromossomo. Se a aproximação não for possível para os pontos em um cromossomo, este é removido da população. Quando a aproximação é realizada com sucesso, são calculadas as distâncias entre cada ponto restante na imagem e a hipótese gerada. Como não existe fórmula fechada para a distância geométrica entre um ponto e uma quádrica, o algoritmo emprega

uma aproximação conhecida como *Pseudo-Distância Geométrica* (PDG). O valor desta aproximação é dado por:

$$PDG = \min\{d_x, d_y, d_z\} \quad (4.11)$$

onde d_x , d_y e d_z são distâncias entre o ponto e a superfície, paralelas aos eixos X , Y , e Z . Segundo os autores, a PDG possui um significado geométrico intrínseco, sendo, portanto, adequada para a utilização em aplicações de engenharia reversa, onde a determinação precisa das formas dos objetos é muito importante. Além disso, a PDG pode ser calculada de forma relativamente fácil, evitando métodos iterativos para o cálculo da verdadeira distância geométrica.

O valor da função de *fitness* de um cromossomo é incrementado para cada ponto cujo valor da PDG é menor que um limiar, de forma semelhante à função de custo do RANSAC, utilizada também nos algoritmos de Roth e Levine, já discutidos.

O AG empregado cria aleatoriamente a população inicial, cujo tamanho, constante durante o processo de evolução, é baseado no número de pontos restantes na imagem. Se há um total de N pontos, o tamanho da população inicial é estabelecido como $\frac{N}{9}$, onde 9 é o número de pontos (genes) em cada cromossomo.

O par de cromossomos de cada reprodução é determinado por uma operação de seleção aleatória. Na operação de *crossover*, cada cromossomo pai é dividido aproximadamente ao meio, resultando em duas partes, uma com quatro e outra com cinco genes. Essas partes são combinadas para gerar dois cromossomos filhos, de modo semelhante ao mostrado na figura 4.7(a). Ressalta-se que esse tipo de operador, que utiliza apenas um ponto de *crossover*, é considerado menos eficiente [Man *et al.*, 1996] que o operador de *crossover* uniforme, mostrado na figura 4.9.

Se o valor de adaptação de um dos filhos resultantes for maior que o valor de adaptação mínimo (VAM) entre todos os indivíduos da população, este filho é inserido na população e o cromossomo apresentando o VAM é removido da população. Caso o valor de adaptação de um filho seja menor que o VAM, este filho sofre uma operação de mutação. Nesta operação, cada gene do cromossomo filho pode ser substituído pelo gene correspondente de um dos pais, de acordo com uma probabilidade P_m pequena.

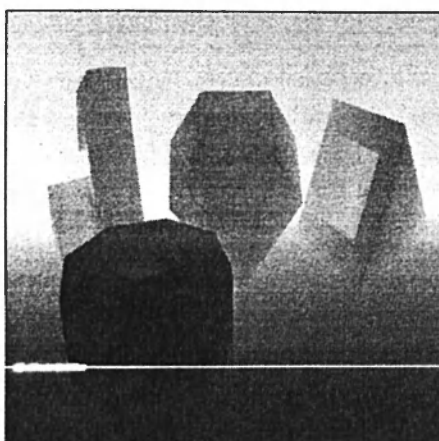
O processo de evolução termina quando o VAM for maior que um valor pré-estabelecido ou quando for atingido um número máximo de iterações, também pré-estabelecido. Ao término, o cromossomo de melhor adaptação é considerado como a superfície extraída e os *inliers* desta superfície são removidos do conjunto de pontos restantes na imagem. Se o número de pontos restantes for maior que um valor pré-determinado, então um novo processo de evolução é iniciado.

Depois que o processo de extração de quádricas termina, as superfícies encontradas são classificadas em tipos específicos de quádricas de modo a fornecer informações mais detalhadas para aplicações de reconstrução de modelos CAD e de reconhecimento de objetos.

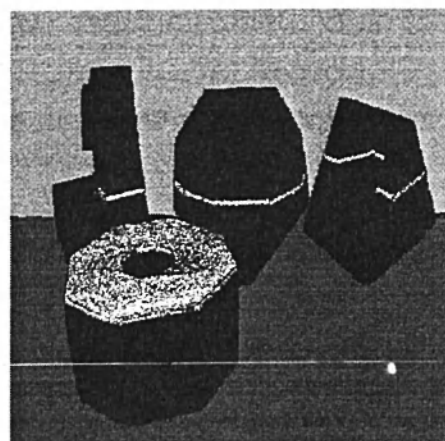
No experimento apresentado em [Chen e Liu, 1999], superfícies quádricas são extraídas com um número surpreendentemente pequeno de iterações do AG (máximo igual a 572 iterações), no entanto, para uma cena contendo poucas superfícies. Espera-se que seja necessário um número maior de iterações na extração de superfícies quádricas de cenas mais complexas. Porém, em uma aplicação de engenharia reversa, para a qual o método foi desenvolvido, o tempo de processamento não é muito importante, sendo de maior relevância a precisão dos parâmetros das superfícies extraídas.

Um ponto negativo é que o método utiliza uma função objetivo semelhante à do RANSAC. Melhores resultados poderiam ser obtidos com a função objetivo do MSAC, sem a introdução de custo computacional adicional.

Outro problema diz respeito ao modo de extração das superfícies. Ao término de cada processo de evolução, os *inliers* identificados para a melhor hipótese de superfície são simplesmente removidos do conjunto de pontos restantes. Contudo, como não estão sendo utilizadas informações de bordas, não se tem nenhuma informação sobre a verdadeira extensão dessa superfície. Assim, muitos pontos de outras superfícies, as quais são interceptadas por essa quádrica de extensão “infinita”, podem ser incorretamente considerados como *inliers*. Isso é ilustrado na figura 4.10(b), na qual os planos correspondentes ao chão e ao fundo, na imagem de profundidade da figura 4.10(a), já foram corretamente extraídos, mas as superfícies nas regiões em preto ainda não. Os pontos mais claros na figura 4.10(b) são os *inliers* identificados durante a extração do terceiro maior plano da imagem. Como a extensão de cada plano não é conhecida, também são incorretamente detectados como *inliers* os pontos nas superfícies dos outros objetos, as quais são interceptadas pelo plano não delimitado. Neste caso, quanto maior a amplitude da faixa de *inliers*, maior o número de pontos incorretamente identificados como *inliers*.



(a) Imagem de profundidade



(b) *Inliers* para terceira superfície extraída

Figura 4.10: Extração de superfície de extensão desconhecida.

4.4.2.3 Yu, Bui, e Krzyzak

Yu *et al.* propoem em [Yu *et al.*, 1994] um algoritmo para segmentação e reconstrução de superfícies de imagens de profundidades, utilizando planos e quádricas como modelos de superfície. O algoritmo emprega um novo método robusto denominado *RESidual Consensus* (RESC), o qual é baseado em histogramas de valores residuais dos pontos da imagem em relação às superfícies hipotéticas, instanciadas a partir de conjuntos mínimos. Um AG também é empregado para acelerar o processo de busca da melhor hipótese.

A definição de valor residual utilizada é a distância algébrica, uma aproximação da verdadeira distância geométrica, dada por:

$$r_i = z_i - \hat{z}_i \quad (4.12)$$

onde z_i é o valor da coordenada z do ponto de índice i , e \hat{z}_i é o valor calculado utilizando a instância (parâmetros) do modelo de superfície.

Primeiramente, é realizada uma segmentação preliminar utilizando um detector de bordas de profundidade. Cada região resultante desta segmentação pode conter várias outras regiões menores, separadas por bordas de orientação. O método RESC é utilizado então para extrair essas regiões internas.

O método RESC é um aperfeiçoamento do método LTS e, como este, durante o processo de otimização, considera tanto o número de *inliers* quanto os resíduos destes. O RESC utiliza um histograma de valores residuais para melhor identificar a parte de resíduos referente aos *inliers* e separá-la da parte de resíduos dos *outliers*, o que é o correspondente à determinação do parâmetro h do método LTS. No entanto, no LTS este parâmetro é determinado previamente e não a partir dos resíduos de cada hipótese de superfície, como acontece no RESC.

Um AG é utilizado para determinar a melhor hipótese de superfície e a região extraída é constituída pelo maior componente conexo formado pelos *inliers* da hipótese final. O AG utilizado é semelhante ao empregado por Roth e Levine: do tipo *steady-state*; com cromossomos representando conjuntos mínimos; operador de *crossover* uniforme; e mutação por alteração aleatória de genes. Por outro lado, a operação de seleção é baseada no *ranking* [Whitley, 1989] de cada cromossomo, e não no valor de adaptação. Este tipo de operador de seleção é capaz de melhor preservar a diversidade genética na população, evitando convergência prematura. O algoritmo também emprega um critério de parada adicional para o processo evolutivo: se a diferença entre o maior e o menor valor de adaptação de cromossomos na população for menor que um valor pré-determinado, então o processo de evolução é interrompido por causa deste indicativo de convergência.

Nos experimentos apresentados, o método RESC foi comparado positivamente em relação ao método LMS. Os experimentos também mostraram que o AG empregado é de 2 a 3 vezes mais rápido que a busca aleatória.

No entanto, o método RESC possui vários parâmetros que devem ser ajustados.

empiricamente. As imagens reconstruídas apresentaram lacunas entre as superfícies aproximadas, o que mostra uma baixa precisão no método em regiões de borda. Outro fator negativo é a utilização da distância algébrica como valor residual, a qual é muito imprecisa, como mostra a figura 4.11.

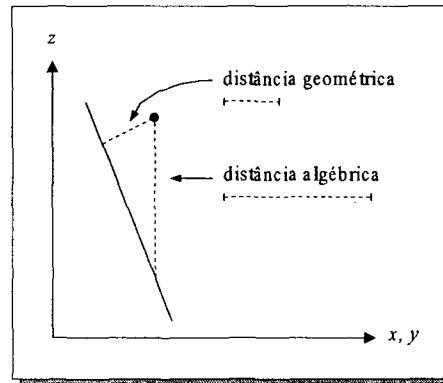


Figura 4.11: Distâncias algébrica e geométrica entre um ponto e um plano.

4.5 Método Desenvolvido

Como resultado do levantamento realizado sobre métodos de aproximação robustos e extração de primitivas geométricas, foi desenvolvido um novo método de segmentação de imagens de profundidades, descrito nesta seção, baseado na extração de superfícies planas.

A escolha do modelo de superfície planar justifica-se pelo fato de que, segundo o que foi apontado no trabalho de Hoover *et al.* [Hoover *et al.*, 1996] e conforme constatado na análise dos trabalhos relacionados (seção 4.4), o problema de segmentação permanece sem uma solução ótima mesmo para imagens contendo apenas superfícies planas. Entre os principais problemas encontrados estão a incorreta identificação das formas e localização das bordas dos objetos, além da não identificação de regiões pequenas da imagem. Ainda sobre o modelo escolhido, observa-se que superfícies de maior ordem podem ser quebradas em partes menores, aproximadamente planas, e estas podem ser posteriormente relacionadas de modo a fornecer informações sobre a real superfície em cena. Porém, o método descrito a seguir não está limitado ao modelo de superfície planar e pode ser estendido de modo a utilizar modelos de superfícies curvas.

O método desenvolvido consiste em um método de segmentação híbrido utilizando técnicas de detecção de bordas e de crescimento de regiões e composto por dois principais estágios de processamento, como mostra a figura 4.12.

O primeiro estágio, o *pré-processamento*, é responsável pelo cálculo de informações inicialmente requeridas pelo segundo estágio. Mais especificamente, ele realiza: (1) o cálculo, local, dos coeficientes do vetor normal das superfícies para cada ponto da imagem; e (2) uma segmentação rústica através da detecção de bordas de profundidades e de orientação.

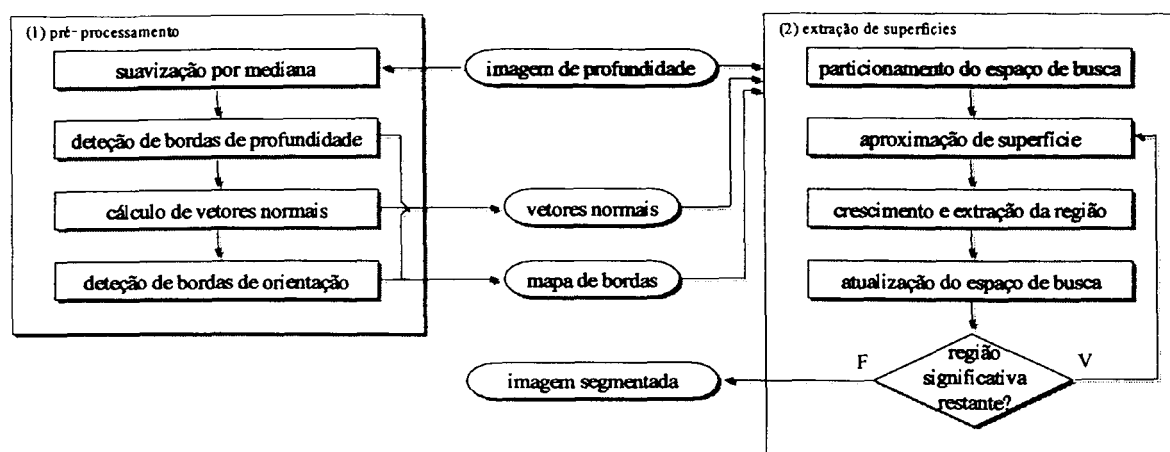


Figura 4.12: Estágios de processamento do método de extração de superfícies.

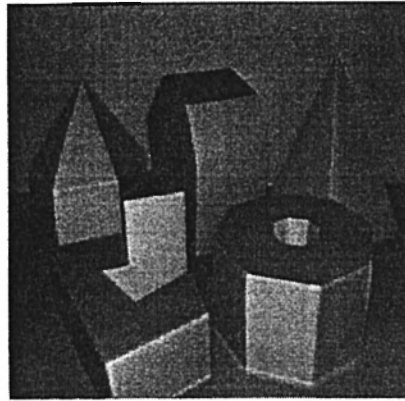
O segundo estágio, a *extração de superfícies*, realiza um processo iterativo de extração de regiões correspondentes a superfícies planas na imagem de profundidade. A cada iteração, uma versão melhorada do método robusto MSAC (seção 4.2.4), a qual utiliza um algoritmo genético aperfeiçoado, é empregada para a identificação dos verdadeiros *inliers* da superfície a ser extraída. O conjunto de pontos identificados como *inliers* é utilizado então como a região semente para um processo de crescimento de regiões realizado com o objetivo de identificar e extrair a região da imagem correspondente à superfície aproximada. Este processo iterativo termina quando não mais restarem regiões de tamanho significativo a serem processadas.

Estes dois estágios principais, acima apresentados, são descritos em maiores detalhes nas seções 4.5.1 e 4.5.2, a seguir. Para auxiliar na descrição dos processos realizados, serão apresentados exemplos de resultados obtidos para a imagem de profundidade da figura 4.13(b), cuja respectiva imagem de intensidade luminosa é mostrada na figura 4.13(a). Esta imagem não é utilizada pelo método e é mostrada aqui apenas para proporcionar ao leitor uma melhor compreensão visual dos objetos em cena. A seção 4.5.3 discute os resultados experimentais e a seção 4.5.4 apresenta uma avaliação quantitativa deste método baseada nos resultados obtidos experimentalmente e utilizando as métricas de Hoover *et al.*

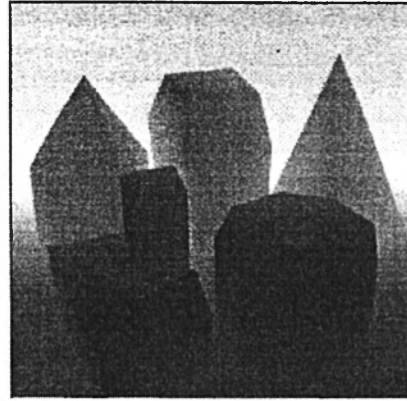
4.5.1 Pré-processamento

Uma razão pela qual este estágio de pré-processamento é primeiramente realizado deve-se ao fato de o método de aproximação robusto utilizado no estágio seguinte requer a disponibilidade prévia de informação de orientação de superfície em cada *pixel* da imagem. Por isso, os coeficientes dos vetores normais em cada *pixel* p da imagem são calculados neste estágio através de uma aproximação LS local de uma superfície plana, utilizando uma máscara de tamanho $N \times N$ centrada em p .

Outro motivo é que os vetores normais obtidos podem ser utilizados para detectar



(a) Imagem de intensidade luminosa



(b) Imagem de profundidade

Figura 4.13: *Imagem de testes Perceptron número 29.*

bordas de orientação, enquanto bordas de profundidade podem ser obtidas dos valores das coordenadas z dos pontos da imagem de entrada. Por isso, este estágio de processamento também produz uma segmentação rústica na forma de uma mapa de bordas, o qual não é imprescindível, mas ajuda a diminuir consideravelmente o custo computacional do processo de extração de superfícies realizado no estágio seguinte. Assim, este primeiro estágio de pré-processamento é baseado no trabalho de segmentação de imagens de profundidade por detecção de bordas desenvolvido por Silva e Bellon [Silva, 2000] [Bellon e Silva, 2002].

Antes da aproximação planar, para atenuar os efeitos de ruído impulsivo e de *pseudo-outliers*, ou seja, para melhorar os coeficientes dos vetores normais estimados, são realizados os seguintes processos: (1) suavização nos valores de coordenada z da imagem utilizando um filtro por mediana de dimensões 3×3 , aplicado uma única vez; e (2) detecção de bordas de profundidade para que, na aproximação planar, quando a máscara for centrada em um *pixel* próximo a uma borda de orientação, os pontos do outro lado da borda possam ser desconsiderados, melhorando a qualidade dos coeficientes obtidos.

Na de detecção de bordas de profundidade, apenas a banda Z da imagem de entrada é utilizada, após sofrer a filtragem por mediana. Este processo é realizado da seguinte forma:

1. Calcula-se uma imagem contendo valores de magnitude de vetores gradiente, conforme mostrado na equação 3.11 (página 46). Neste cálculo, os valores das duas primeiras derivadas parciais, f_x e f_y , em cada *pixel* p são aproximados pelos valores absolutos das diferenças nos valores da coordenada z , respectivamente, entre os *pixels* imediatamente à direita e à esquerda de p , e entre aqueles imediatamente acima e abaixo de p . Os valores de magnitude de gradiente obtidos são maiores nas regiões de bordas de profundidade.
2. Calcula-se um limiar automático t_p baseado nos valores de média e desvio padrão

dos valores de magnitude de gradiente [Haddon, 1988].

3. Considera-se, então, como *pixels* de borda de profundidade todos aqueles apresentando valores de magnitude de gradiente maiores que o valor de limiar calculado.

Após esse processo, realiza-se então o cálculo dos vetores normais através da aproximação local de um plano. Para cada *pixel* p da imagem, uma máscara de dimensões $N \times N$ é centrada em p e os demais *pixels* na região coberta pela máscara são classificados como *outliers* se estiverem de um lado diferente do qual p se encontra em relação a uma das bordas de profundidade detectadas. Esses *outliers* são descartados e as coordenadas (x, y, z) dos demais *pixels* são utilizadas na aproximação, a qual emprega o método de *Householder* [Press et al., 1992] para identificar o auto-vetor associado ao menor autovalor da matriz *scatter* 3×3 calculada a partir do conjunto de pontos utilizado. Este auto-vetor é considerado então o vetor normal estimado.

A detecção de bordas de orientação é realizada de forma análoga à de bordas de profundidade, utilizando, no entanto, o gradiente de desvio angular entre os vetores normais de *pixels* vizinhos a um pixel p . Um limiar automático t_o é calculado e utilizado na identificação dos *pixels* considerados como borda de orientação.

Um mapa de bordas é formado pela sobreposição das bordas de profundidade e de orientação detectadas. A figura 4.14(a) mostra o mapa de bordas obtido para a imagem de profundidade da figura 4.13(b). Nota-se que as bordas de profundidades detectadas estão bem localizadas e descrevem corretamente parte do contorno dos objetos. Já as bordas de orientação neste mapa de bordas apresentam muitas vezes formas distorcidas, sendo que algumas delas estão incompletas e outras não foram detectadas. Além disso, o mapa apresenta bordas com 2 *pixels* de largura. Mas como a finalidade do mapa de bordas calculado é representar apenas uma segmentação rústica, intermediária, estes problemas não apresentam grande relevância. Também não são requeridos processos de afinamento ou fechamento de bordas.

O resultado final deste estágio de processamento é constituído pela imagem contendo os coeficientes dos vetores normais calculados para cada *pixel* e pela segmentação inicial na forma de um mapa de bordas.

4.5.2 Extração de Superfícies

Os dados de entrada deste estágio de processamento são constituídos pela imagem de profundidade original (não suavizada), além da imagem de vetores normais e do mapa de bordas calculados no estágio anterior.

Cada um dos principais procedimentos realizados neste estágio são apresentados em detalhes a seguir. O primeiro deles, realizado antes do início do processo iterativo de extração de regiões de superfície plana, é o particionamento do espaço de busca, descrito abaixo.

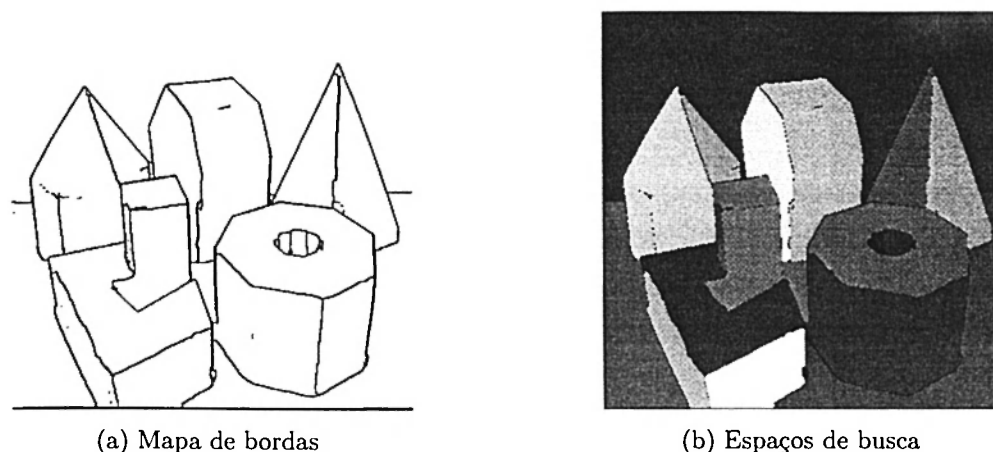


Figura 4.14: Segmentação rústica inicial para determinação de espaços de busca.

4.5.2.1 Particionamento do Espaço de Busca

O principal objetivo deste procedimento é dividir os dados da imagem em conjuntos de pontos menores, cada um descrevendo um número pequeno de superfícies a serem extraídas. Ou seja, a finalidade deste procedimento é particionar o espaço de busca, a imagem de profundidade, antes de iniciar a processo de extração de superfícies planas para cada subconjunto em separado.

Este não é um procedimento indispensável ao método de segmentação desenvolvido, o qual pode processar todos os pontos na imagem como sendo o único espaço de busca por superfícies a serem extraídas. Contudo, o particionamento do espaço de busca aqui realizado contribui consideravelmente para a redução do custo computacional do método de segmentação. O método de aproximação robusto descrito na próxima seção pode considerar um número menor de hipóteses de superfície e, além disso, não é necessário o cálculo de valores residuais para a grande quantidade de pontos fora do espaço de busca sendo considerado. Ainda, como, em cada espaço de busca, o número de superfícies é menor, a porcentagem de *inliers* para cada superfície é relativamente maior e, por isso, maior é a precisão dos parâmetros estimados para cada superfície.

A determinação de um conjunto de espaços de busca é realizada através de um crescimento de regiões conexas (*flood fill*) [Sonka *et al.*, 1998] no mapa de bordas calculado anteriormente. Neste processo, cada ponto imediatamente acima, abaixo, à esquerda ou à direita (vizinhança 4) de um ponto da região em crescimento é adicionado a ela através da atribuição de um mesmo rótulo de região. O crescimento de cada região é limitado pelas bordas detectadas. Como estas bordas possuem largura de dois pontos, é permitido a cada região crescer um ponto sobre a largura da borda. Deste modo, a maioria dos pontos de uma borda acaba sendo distribuída entre as duas regiões separadas pela borda. A imagem de regiões correspondente ao mapa de bordas da figura 4.14(a) é mostrada na figura 4.14(b).

Cada região obtida é considerada então como um espaço de busca a ser processado

para extração de superfícies planas. Espaços muito pequenos, com quantidades de pontos menores que uma pequena porcentagem t_e (e.g. $t_e = 0,03\%$), pré-especificada, do número total de pontos na imagem de profundidade são desconsiderados.

Após a determinação do conjunto de espaços de busca inicial CEB_0 , inicia-se o processo iterativo de extração de superfícies, cujos três principais procedimentos são descritos nas seções a seguir. A cada iteração, o maior espaço de busca E_i em CEB_i é processado e a maior região plana em E_i é extraída. As regiões conexas de pontos restantes em E_i são então identificadas e inseridas como novos espaços de busca em CEB_{i+1} , caso o número de pontos em cada uma seja maior que a porcentagem mínima pré-estabelecida. Este processo termina quando não restarem mais espaços de busca a serem processados, ou seja, quando $CEB_{i+1} = \phi$. A imagem de saída, o resultado da segmentação, é composta pelas regiões planas extraídas em cada iteração.

4.5.2.2 Aproximação de Superfície

Esta etapa de processamento é responsável pelo cálculo dos coeficientes da maior superfície plana descrita por pontos no espaço de busca E_i . Outra tarefa realizada nesta etapa é a identificação do conjunto de pontos *inliers* da superfície estimada, os quais são posteriormente utilizados como sementes para o processo de crescimento de superfície descrito na próxima seção. Para alcançar estes objetivos, esta etapa de aproximação de superfícies utiliza um novo método robusto, o qual foi desenvolvido como um aperfeiçoamento do método MSAC e emprega um algoritmo genético com um novo conjunto de parâmetros para acelerar seu processo de otimização.

Um problema encontrado na extração de superfícies é que, na maioria dos casos, não se tem informações completas sobre a real extensão de cada superfície sendo extraída. Ou seja, não se tem o contorno completo e fechado da região correspondente à superfície. Por isso, pontos de outras superfícies interceptadas por essa superfície de extensão “infinita” podem ser classificados incorretamente como *inliers*. Como já foi mostrado na figura 4.10(b) (página 85), estes *pseudo-inliers* podem ser pontos nas superfícies de outros objetos. Uma possível solução para esse caso poderia consistir em considerar apenas o maior componente conexo de *inliers* e descartar o demais. No entanto, isso pode gerar sobre-segmentação caso os verdadeiros *inliers* da superfície não constituam uma única região conexa. Além disso, *pseudo-inliers* podem ser pontos de outra superfície do mesmo objeto, os quais foram incorretamente detectados devido a lacunas ou mesmo falha na detecção da borda de orientação separando duas superfícies do objeto. Este segundo caso, que não pode ser resolvido através da análise de regiões conexas, também pode causar sobre-segmentação, mas de uma superfície vizinha, como mostra a figura 4.15. A figura 4.15(c) mostra os *inliers*, identificados utilizando o método MSAC, para o plano da região 1 da figura 4.15(b), a qual por sua vez corresponde à segmentação manual das superfícies na figura 4.15(a). Neste exemplo, os *pseudo-inliers*, que também seriam detectados por

outros métodos robustos que não o MSAC, foram responsáveis pela sobre-segmentação da região 2.

Pseudo-inliers exercem influência no valor da função de custo de métodos robustos pois são considerados como verdadeiros *inliers* e, deste modo, podem causar imprecisões nos parâmetros de superfície estimados. Assim, para se obter uma aproximação de superfície mais precisa e evitar sobre-segmentação, é preciso classificar *pseudo-inliers* como *outliers* para que estes não exerçam influência na função de custo do método robusto.

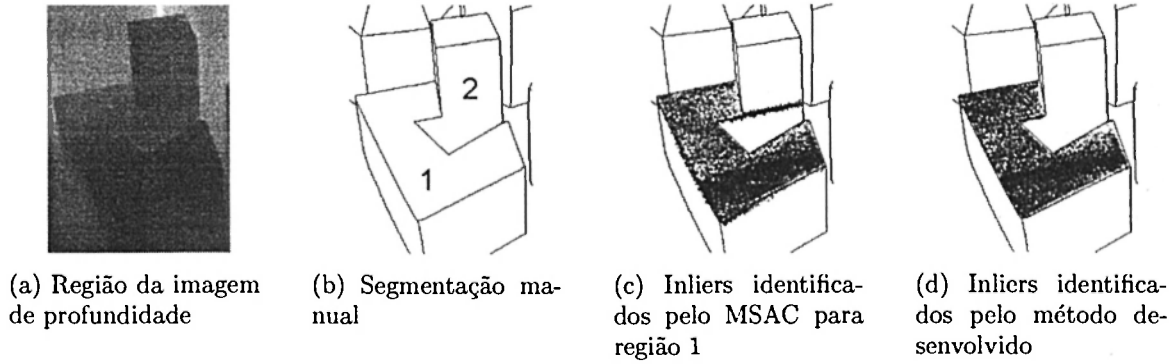


Figura 4.15: Correta classificação de pseudo-inliers como outliers.

A abordagem escolhida para evitar os problemas causados por *pseudo-inliers* consiste na adição de informação sobre orientação de superfície à função de custo do MSAC e no conceito de *ponto compatível*. Um ponto p pode ser considerado “compatível” com uma hipótese de superfície caso esteja de acordo com as seguintes restrições:

1. p está suficientemente próximo à hipótese de superfície. Ou seja, seu valor residual r_p é menor que um limiar de faixa de *inliers* σ .
2. A orientação de superfície estimada localmente no ponto p está de acordo com a orientação apresentada, em p , pela superfície hipotética. Isto é, o ângulo formado entre o vetor normal calculado localmente em p ($\vec{n}_{l,p}$) e o vetor normal da hipótese de superfície no ponto p ($\vec{n}_{h,p}$) deve ser menor que um limiar θ .

A função de custo do método robusto resultante considera como *inliers* apenas os pontos compatíveis com a hipótese de superfície sendo avaliada. Esta função é definida em termos dos valores residuais r_1, \dots, r_n como

$$f(r_1, \dots, r_n) = \sum_{j=1}^n \rho(r_j) \quad (4.13)$$

onde o novo termo residual robusto $\rho()$ é dado por

$$\rho(r) = \begin{cases} r & \text{se } r < \sigma \text{ e } \cos^{-1}(\vec{n}_{l,j} \cdot \vec{n}_{h,j}) \leq \theta \\ \sigma & \text{caso contrário} \end{cases} \quad (4.14)$$

Uma possível maneira de estabelecer o valor do limiar θ é defini-lo em termos do limiar t_o , calculado automaticamente durante a detecção de bordas de orientação (e.g. $\theta = \frac{t_o}{3}$). Assim, no exemplo da figura 4.15, como os planos das regiões 1 e 2 são perpendiculares, os vetores normais calculados localmente para os *pseudo-inliers* na região 2 não estão de acordo com o vetor normal da hipótese de plano para a região 1. Deste modo, os *pseudo-inliers* são classificados como *outliers* e o resultado é mostrado na figura 4.15(d), na qual a sobre-segmentação da região 2 foi evitada.

Os coeficientes (a, b, c) de cada hipótese de superfície plana são obtidos a partir de três pontos quaisquer, p_1 , p_2 e p_3 , em E_i e do seguinte produto vetorial:

$$\vec{n}_h = (a, b, c) = \vec{u} \times \vec{v} \quad (4.15)$$

onde os vetores \vec{u} e \vec{v} são definidos como

$$\vec{u} = (u_1, u_2, u_3) = p_2 - p_1 \quad \vec{v} = (v_1, v_2, v_3) = p_3 - p_1 \quad (4.16)$$

e o produto vetorial é calculado através da expansão do determinante

$$\begin{vmatrix} i & j & k \\ u_1 & u_2 & u_3 \\ v_1 & v_2 & v_3 \end{vmatrix} = ai + bj + ck \quad (4.17)$$

e pela coleta dos termos em função de i , j e k . O coeficiente d é então calculado substituindo-se os coeficientes (a, b, c) e as coordenadas (x_p, y_p, z_p) de um dos pontos p_1 , p_2 ou p_3 na equação implícita do plano e resolvendo para d :

$$d = -ax_p - by_p - cz_p \quad (4.18)$$

Para que as hipóteses de superfícies planas apresentem vetores normais unitários e sempre apontando no mesmo sentido (para fora ou para dentro dos objetos) os coeficientes obtidos são primeiramente normalizados (divididos por $\sqrt{a^2 + b^2 + c^2}$) e em seguida podem ter os sinais invertidos (podem ser multiplicados por -1) para que os coeficientes c de todas as hipóteses apresentem sempre o mesmo sinal.

A partir dos coeficientes normalizados, valores residuais correspondentes às verdadeiras distâncias geométricas, r_j , entre cada um dos demais pontos p_j , $0 \leq j \leq n$, em E_i e uma hipótese de plano com coeficientes (a_h, b_h, c_h, d_h) podem ser obtidas de forma bastante simples:

$$r_j = |a_h x_j + b_h y_j + c_h z_j + d_h| \quad (4.19)$$

O algoritmo genético utilizado para acelerar o processo de otimização das hipóteses de superfície é um AG do tipo *steady-state* e contém uma população de cromossomos constituídos, cada um, por três genes correspondentes a pontos do espaço de busca E_i , os

quais são utilizados para instanciar os parâmetros de um plano conforme descrito acima. A cada cromossomo é atribuído um valor de adaptação (*fitness*) inversamente proporcional, recíproco, ao valor de custo obtido com a função de estimativa robusta da equação 4.13. Assim, ao maximizar o valor de adaptação, o AG minimiza o custo da melhor hipótese de superfície.

Os cromossomos da população inicial são formados por pontos escolhidos aleatoriamente em E_i . A cada ciclo de reprodução, dois novos cromossomos são criados e inseridos na população. Para manter o tamanho da população fixo, o dois cromossomos de menor adaptação são descartados. O critério de parada utilizado no processo evolutivo consiste em um numero fixo de iterações de reprodução, t_I .

Para diminuir a possibilidade de convergência precoce, ou seja, para manter a variedade genética na população, não é permitida a existência de cromossomos idênticos sendo que um dos cromossomos em duplicata é descartado. Além disso, utiliza-se uma operação de seleção por torneio, mais eficiente que a seleção proporcional pois evita o surgimento precoce de uma “elite” de cromossomos [Zhang e Kim, 2000]. Na seleção por torneio, cada cromossomo selecionado para reprodução é definido como o cromossomo de melhor adaptação entre dois cromossomos escolhidos aleatoriamente na população.

O operador de *crossover* utilizado é um operador uniforme (figura 4.9, página 82) que apresenta uma grande capacidade de promover diversidade genética. A operação de *crossover* é realizada em todas as iterações do ciclo evolutivo, ou seja, $P_c = 1,0$. Após a operação de *crossover*, cada cromossomo resultante pode sofrer mutação de acordo com uma probabilidade P_m .

Como a mutação por alteração aleatória é pouco eficiente na escolha de genes “bons” para um cromossomo, já que na alteração de um gene ela não leva em consideração seu valor atual, uma outra operação de mutação foi desenvolvida. Nesta operação, o ponto em cada gene sofrendo mutação é deslocado um número Δ_m de *pixels* em uma direção horizontal (esquerda ou direita) ou vertical (acima ou abaixo), como ilustrado na figura 4.16(a). Direção e sentido de deslocamento são determinadas de forma aleatória e com a mesma probabilidade. Se o deslocamento em um sentido ultrapassar os limites do espaço (região) de busca E_i , então o sentido é invertido (figura 4.16(b)). Se o deslocamento em ambos os sentidos em uma direção não for possível, a outra direção é testada (figura 4.16(c)). A idéia por trás desta operação é melhorar os genes contendo pontos em regiões de ruído, bordas ou próximos aos limites do espaço de busca, movendo-o para regiões mais confiáveis e adequadas, como por exemplo a mesma superfície dos outros pontos. O valor de deslocamento Δ_m é influenciado pelas dimensões da imagem (matriz) e deve ser determinado experimentalmente. Nos experimentos realizados, este operador proporcionou melhores resultados que o operador de mutação aleatória.

Os parâmetros previamente requeridos pelo AG são: (1) o tamanho da população, t_{pop} ; (2) o probabilidade de mutação, P_m ; (3) o valor de deslocamento, Δ_m utilizado na mutação; e (4) o número de iterações do ciclo evolutivo, t_I . A determinação destes

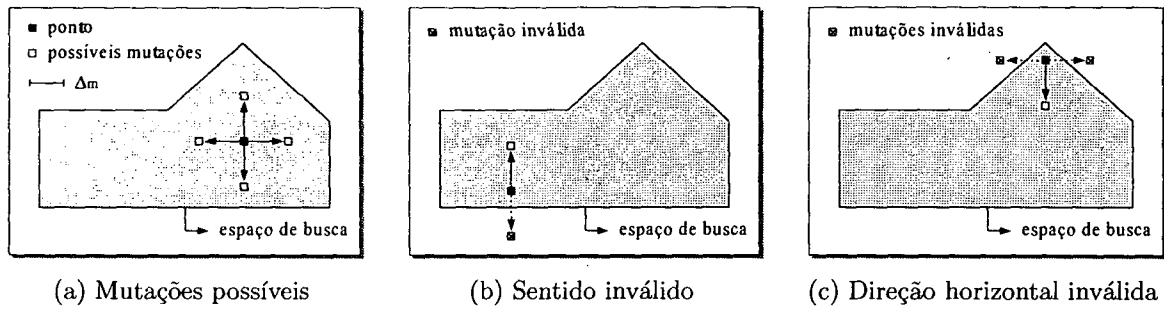


Figura 4.16: *Mutação por deslocamento do ponto dentro do espaço de busca.*

parâmetros é discutida na seção 4.5.3.

Ao término do processo evolutivo, os pontos do cromossomo de melhor adaptação são utilizados para instanciar a melhor hipótese de superfície plana determinada pelo AG. Os *inliers* desta superfície são então utilizados em uma aproximação LS final – pelo método de *Householder*, também utilizado na seção 4.5.1 – para melhorar a precisão dos coeficientes da superfície.

4.5.2.3 Crescimento e Extração da Região

Em imagens de profundidade reais, muitos dos pontos da superfície a ser extraída podem não ser classificados como *inliers* por causa de efeitos de ruído. O conjunto de *inliers* identificado na etapa anterior pode até mesmo constituir-se de uma nuvem esparsa de pontos não correspondendo a uma única região conexa. Para evitar sobre-segmentação e identificar todos os pontos da região a ser extraída, os *inliers* identificados no processo de aproximação de superfície são utilizados como sementes de um processo de crescimento de regiões.

Este processo de crescimento agrupa aos *inliers* sementes todos os seus pontos vizinhos que sejam pontos compatíveis com a superfície aproximada. Cada ponto anexado à região de *inliers* é considerado então como uma nova semente. O crescimento termina quando mais nenhum ponto vizinho puder ser anexado. Neste processo, o grau de tolerância para se considerar um ponto como compatível é maior que o utilizado na etapa anterior. Ou seja, os valores dos limiares σ_c e θ_c utilizados no crescimento da região são maiores que os valores dos limiares σ e θ empregados na aproximação de superfície. Uma explicação para isso é que, durante a aproximação, o grau de tolerância para classificação de *inliers* deve ser pequeno para evitar desvios na superfície aproximada e melhorar a precisão dos coeficientes estimados. Assim, os limiares utilizados nesta etapa são definidos em termos dos limiares pré-especificados para a etapa anterior:

$$\theta_c = C\theta \quad \sigma_c = C\sigma \quad (4.20)$$

onde C é uma constante pequena e tipicamente $1,0 < C < 5,0$.

O maior componente resultante do processo de crescimento é considerado como a região R_i correspondente à superfície aproximada. Todos os pontos em R_i são rotulados com o valor i e são copiados para a imagem de saída, a qual contém todas as superfícies extraídas até o momento e , no final do estágio de extração, constituíra o resultado da segmentação.

Durante o crescimento, já que as bordas de profundidade detectadas no estágio de pré-processamento estão confiavelmente localizadas, elas são utilizadas para limitar o processo de crescimento em uma determinada direção. Já que essas bordas apresentam dois *pixels* de largura, é permitido ao processo crescer um *pixel* sobre a largura de cada borda. No entanto, os *pixels* de borda anexados não são considerados como novas sementes, o que interrompe o crescimento na direção perpendicular à borda.

Diferentemente, as bordas de orientação detectadas no estágio de pré-processamento são úteis apenas para o particionamento do espaço de busca, reduzindo o custo computacional da etapa de aproximação de superfície. Como visto na seção 4.5.1, estas bordas nem sempre estão corretamente localizadas e, por isso, não podem ser utilizadas para limitar o processo de crescimento. Deste modo, é permitido ao processo de crescimento ultrapassar os limites do espaço de busca, pois estes limites podem corresponder a bordas de orientação incorretamente localizadas e alguns pontos da superfície sendo extraída podem estar do outro lado dessas bordas.

Como não existe informação sobre bordas de orientação para limitar o processo de crescimento, as regiões extraídas a cada iteração acabam agregando também alguns pontos de regiões vizinhas, como mostra a figura 4.17(a). Isso ocorre porque, pontos próximos a, e nos dois lados de, uma borda de orientação acabam sendo considerados pontos compatíveis com ambas as superfícies separadas pela borda. As razões para isso são: (1) conforme ilustrado pela área em cinza na figura 4.17(b), alguns pontos estão localizados dentro da faixa de *inliers* de ambas as superfícies; e (2) os vetores normais aproximados localmente podem apresentar uma orientação intermediária quando calculados em regiões de bordas de orientação pelo fato de pontos de diferentes superfícies serem utilizados na aproximação LS. Assim, esses vetores normais acabam estando de acordo com as orientações de ambas as superfícies separadas pela borda.

É necessário então estabelecer um critério na análise destes pontos ambíguos para corretamente segmentar as regiões e preservar a localização das bordas de orientação dos objetos em cena. A figura 4.17(b) mostra que o critério no qual um ponto é atribuído à superfície para a qual está mais próximo é falho pois, neste exemplo, classifica incorretamente os pontos evidenciados pelos três círculos.

Um critério apropriado é o desenvolvido por Fitzgibbon *et al.* [Fitzgibbon *et al.*, 1997] no qual um ponto é anexado à região localizada no mesmo lado de uma *superfície de decisão*. No caso de dois planos, esta superfície é um outro plano contendo a linha de interseção dos dois planos originais e dividindo ao meio o volume de espaço entre eles.

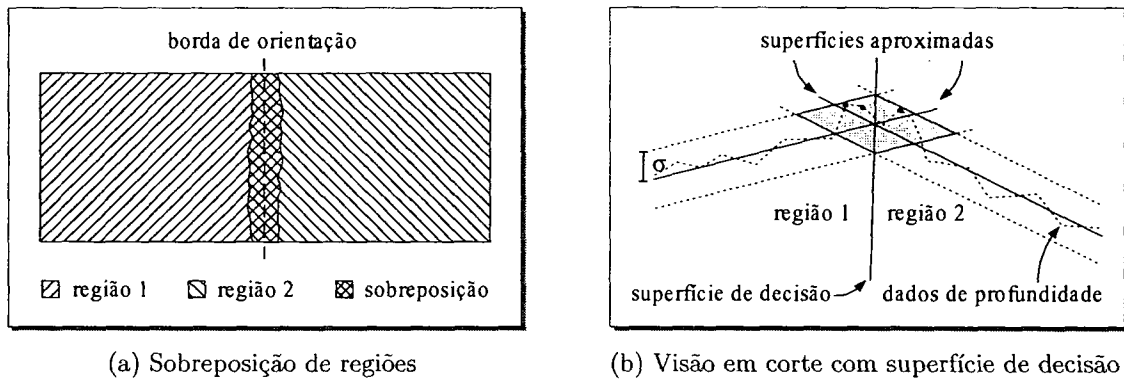


Figura 4.17: Crescimento de superfície em região de borda de orientação.

(figura 4.17(b)). Para dois planos

$$a_1x + b_1y + c_1z + d_1 = 0 \quad a_2x + b_2y + c_2z + d_2 = 0 \quad (4.21)$$

o plano de decisão de dado por

$$a_3x + b_3y + c_3z + d_3 = (a_1 - a_2)x + (b_1 - b_2)y + (c_1 - c_2)z + (d_1 - d_2) = 0 \quad (4.22)$$

A equação do plano de decisão é utilizada então para resolver o problema do crescimento de regiões próximo a bordas de orientação. O lado em que cada superfície se encontra em relação ao plano de decisão é representado pelo sinal do produto interno entre o vetor normal da superfície (\vec{n}_s) e o do plano de decisão (\vec{n}_d):

$$\vec{n}_s \cdot \vec{n}_d = a_s a_d + b_s b_d + c_s c_d \quad (4.23)$$

Assim, considerando os sinais das distâncias geométricas dos pontos na região de sobreposição, aqueles que apresentem distâncias positivas em relação ao plano de decisão são anexados à região cujo vetor normal da superfície produz um valor positivo no produto interno acima. Os demais pontos são anexados à outra região.

Uma última observação sobre este critério é que ele pode ser utilizado também com superfícies curvas se forem consideradas as interseções de planos tangentes, calculados com base no vetor normal em cada ponto.

4.5.2.4 Atualização do Espaço de Busca

Após a região correspondente à superfície aproximada ser identificada e copiada para a imagem de saída, é necessário atualizar o espaço de busca E_i , desconsiderando os pontos da região extraída, R_i .

Um crescimento de regiões conexas é realizado então no conjunto de pontos $E_i - R_i$ produzindo m subconjuntos de pontos conexos S_j , $1 \leq j \leq m$. Cada subconjunto S_j contendo um número de pontos menor que o limiar de tamanho mínimo t_e (seção 4.5.2.1).

é desconsiderado. Os demais subconjuntos são considerados como os novos espaços de busca resultantes da extração da região R_i e são adicionados ao conjunto de espaços de busca (CEB) a ser utilizado na próxima iteração:

$$CEB_{i+1} = (CEB_i - E_i) \cup S, \quad S = \{S_j / |S_j| > t_e\} \quad (4.24)$$

Atualizado o conjunto de espaços de busca, realiza-se o teste de parada que consiste em verificar se ainda existem espaços de busca (regiões) a serem processadas ($CEB_{i+1} \neq \phi$). Caso verdadeiro, o maior espaço de busca em CEB_{i+1} é escolhido como E_{i+1} e uma nova iteração é iniciada.

Quando o último espaço de busca é processado e todos os subconjuntos S_j resultantes são descartados por apresentarem tamanhos não significativos, o processo iterativo é interrompido e o estágio de extração de superfícies termina. A segmentação da imagem de profundidade é representada pela imagem de saída contendo todas as regiões extraídas.

4.5.3 Resultados Experimentais

Nesta seção são discutidos os resultados experimentais obtidos com o método de segmentação por extração de superfícies, o qual também foi desenvolvido em C e C++ e utilizando o ambiente KHOROS. A implementação do AG foi baseada no código fonte da biblioteca *PGAPack*, disponível em <ftp://ftp.mcs.anl.gov/pub/pgapack/pgapack.tar.Z>. O método foi aplicado a imagens da base Perceptron e os resultados foram utilizados na avaliação quantitativa, apresentada na seção 4.5.4.

Recapitulando, o método desenvolvido requer que os seguintes parâmetros em suas respectivas etapas de processamento:

- O estágio de *pré-processamento* requer que o parâmetro N , o tamanho da máscara utilizada no cálculo dos vetores normais, seja previamente determinando. Os limiares t_p e t_o utilizados, respectivamente, na detecção de bordas de profundidade e de orientação são calculados automaticamente.
- As etapas de *particionamento* e de *atualização do espaço de busca* utilizam o limiar t_e para identificar quais regiões não apresentam tamanho significativo e podem ser descartadas.
- Na etapa de *aproximação de superfícies* são necessários os limiares da faixa de *inliers* σ e de desvio angular θ , utilizados pelo método robusto na identificação de pontos compatíveis. Os parâmetros previamente requeridos pelo AG são: o tamanho da população (t_{pop}), a probabilidade de mutação (P_m), o valor de deslocamento da mutação (Δ_m) e o número de iterações do processo evolutivo (t_I).
- A etapa de *crescimento e extração da região* requer apenas o valor da constante C para definir os limiares σ_c e θ_c em função dos limiares σ e θ da etapa anterior.

A determinação dos parâmetros N e t_e é influenciada principalmente pelo tamanho da imagem de entrada. A máscara de aproximação local de dimensões $N \times N$ deve ser grande o suficiente para conter uma quantidade adequada de pontos para que os coeficientes dos vetores normais sejam calculados de forma relativamente precisa na presença de ruídos. No entanto, se ela for muito grande, os coeficientes calculados para regiões pequenas da imagem são muito imprecisos. Nos experimentos realizados com as imagens Perceptron, de dimensões 512×512 *pixels*, obteve-se um melhor resultado quanto à forma das regiões extraídas para o valor de $N = 15$ (i.e. utilizando uma máscara de aproximação de tamanho 15×15).

Em relação ao valor determinado para o limiar t_e , foram observados os seguintes fatos:

1. Valores muito pequenos para este limiar causaram um aumento na ocorrência de sobre-segmentação. O motivo para isso é que pequenas regiões, ruidosas, acabam não sendo extraídas junto com suas respectivas superfícies. Quando estas regiões de ruído apresentam um tamanho maior que t_e , elas são considerados como espaços de busca válidos e os planos extraídos não correspondem a nenhuma das superfícies em cena.
2. Por outro lado, valores grandes para este limiar fazem com que regiões pequenas, mas verdadeiramente correspondentes a superfícies pequenas dos objetos, não sejam processadas pelo método. No resultado da segmentação, estas regiões “perdidas” podem permanecer sem rótulo de superfície ou então serem anexadas no processo de crescimento de uma das superfícies vizinhas.

Para equilibrar estes dois fatos, estabeleceu-se empiricamente um valor $t_e = 121$ *pixels*, o que corresponde a 0,055% do total de *pixels na imagem* e cujo tamanho é equivalente ao de uma região quadrada de dimensões 11×11 . Além do pequeno tamanho relativo, outro fato ressaltado é que os coeficientes dos vetores normais calculados, utilizando uma máscara de dimensões 15×15 (225 *pixels*), para regiões de tamanhos menores que este são muito imprecisos.

Para se estabelecer o conjunto de valores para os parâmetros requeridos pelo AG, foi realizado o experimento descrito a seguir. Considerou-se toda a imagem da figura 4.18(a) como um único espaço de busca no qual o método robusto e o AG desenvolvidos foram aplicados para identificar os *inliers* da maior superfície planar, neste caso o plano correspondente ao chão da cena na imagem, conforme ilustrado na figura 4.18(b). A região deste plano é ligeiramente maior que a região do plano de fundo da imagem. Cada uma dessas superfícies ruidosas contém aproximadamente 30% do total de pontos. Nesse contexto, queria-se identificar uma superfície em um conjunto de pontos apresentando menos de 30% de *inliers*, o que acontece com razoável frequência, sendo que a identificação do plano de fundo, figura 4.18(c), corresponderia a ocorrência de convergência precoce na minimização da função de custo do método robusto.



Figura 4.18: *Imagens do experimento de determinação dos parâmetros do AG.*

Após alguns testes, estabeleceu-se os limiares do método robusto como $\sigma = 2,0$ e $\theta = 30$ para dar suporte ao experimento de definição dos parâmetros do AG, para os quais foram definidos conjuntos de valores potencialmente adequados:

$$\begin{aligned}
 t_{pop} &\in S_{pop} = \{30, 50, 70, 90, 110\} \\
 P_m &\in S_{pm} = \{0.01, 0.05, 0.10, 0.15, 0.20\} \\
 \Delta_m &\in S_{\Delta} = \{3, 5, 7, 9, 11\}
 \end{aligned} \tag{4.25}$$

O número de iterações foi estabelecido como $t_I = 1000$, cerca de duas vezes o número de hipóteses que seriam necessárias na busca aleatória para garantir, com aproximadamente 100% de confiabilidade e 20% de *inliers*, uma hipótese formada apenas por pontos “bons”.

Para cada conjunto de parâmetros p_i , $p_i \in S_{pop} \times S_{pm} \times S_{\Delta}$, foram realizados 100 testes e foi analisada a média dos menores valores obtidos para a função de custo em cada teste. O objetivo era não só identificar o melhor conjunto de parâmetros – os quais melhor aceleravam a busca, sem causar convergência precoce – mas também o número de iterações a partir do qual não ocorriam alterações significativas no valor da função de custo. O conjunto que apresentou a menor média para o valor de custo final foi:

$$t_{pop} = 70 \quad P_m = 0.05 \quad \Delta_m = 9 \tag{4.26}$$

A figura 4.19 mostra o gráfico da evolução média dos valores de custo obtidos nos 100 testes realizados para o método utilizando (1) amostragem aleatória, (2) o AG desenvolvido, com o conjunto de parâmetros acima, e (3) um AG convencional, utilizando os mesmos valores dos limiares t_{pop} e P_m acima, mas com operação de seleção proporcional e mutação aleatória. Como a criação da população inicial dos AGs é feita aleatoriamente e equivale, assim, à busca aleatória, as evoluções são mostras a partir da avaliação da função de custo de número 70, que é o tamanho da população dos AGs.

No gráfico, nota-se que o AG desenvolvido necessitou de aproximadamente 275 avaliações para obter o mesmo valor de custo obtido na busca aleatória após 700. Esse

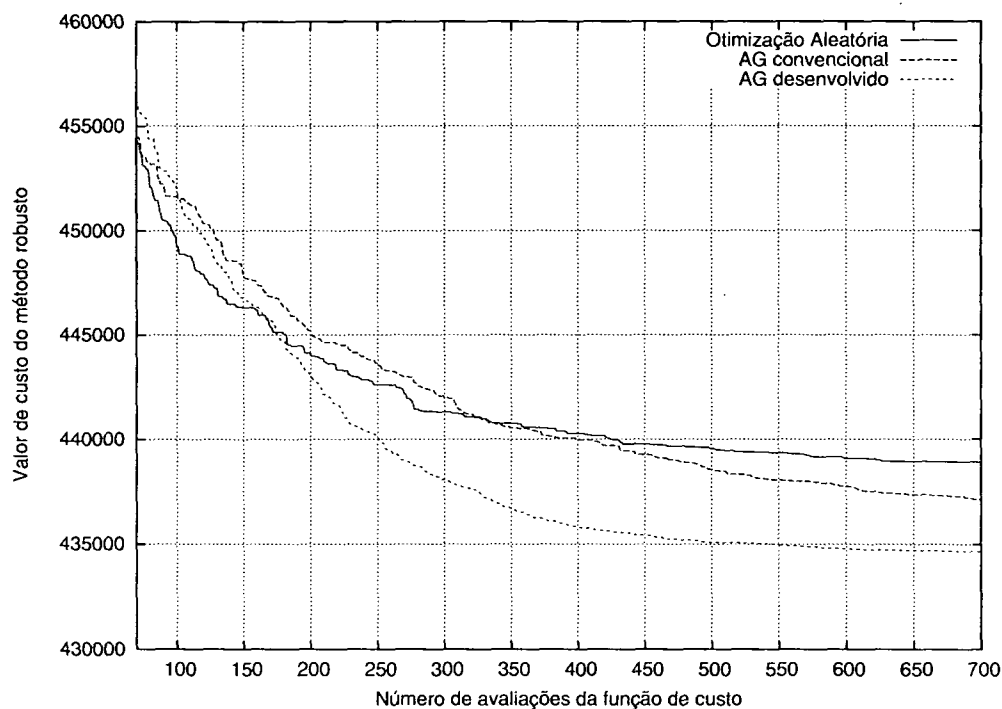


Figura 4.19: *Evolução média dos valores da função de custo do método robusto utilizando diferentes processos de otimização.*

valor de custo foi obtido pelo AG convencional somente após aproximadamente 475 avaliações. Além disso, o AG desenvolvido precisou de menos de 350 avaliações para obter um valor de custo melhor que aquele apresentado pelo AG convencional após 700. A busca aleatória teve melhor desempenho médio nas primeiras 175 avaliações, antes de ser ultrapassada pelo AG desenvolvido, e aproximadamente a partir da avaliação de número 325 apresentou valores de custo maiores que os obtidos pelo AG convencional.

Contudo, estes dados não podem ser analisados isoladamente pois uma maior aceleração da busca pode causar convergência precoce, o que neste caso corresponde à identificação do plano de fundo da imagem. No caso da busca aleatória, isso ocorreu em 36% dos testes realizados. Para o AG convencional, obteve-se um índice de 20% de convergência precoce. Interessantemente, o AG desenvolvido além de acelerar o processo de otimização também apresentou menor índice de convergência precoce, de apenas 13%. Ou seja, este AG foi mais eficiente na busca pelo mínimo global, o que resultou nos menores valores médios dos custos obtidos. O índice obtido pode ser considerado pequeno em relação à grande semelhança nos tamanhos e custos apresentados pelas duas maiores regiões da imagem.

Inicialmente, até a iteração de número 100, o AG desenvolvido apresentou custos maiores, possivelmente por apresentar maior diversidade genética, e no final foi o único a apresentar valores menores que 350000. Nos experimentos realizados, verificou-se que este

valor de custo já é adequado para a identificação de *inliers* realizada antes da aproximação LS. Além disso, nos intervalos entre as interações 500–550, 550–600, 600–650 e 650–700 não houve ganho considerável nos valores de custo. Desta forma, nos demais experimentos realizados, estabeleceu-se o valor 500 para o número necessário de avaliações da função de custo, sendo 70 destas avaliações correspondentes à criação da população inicial e as outras $t_I = 430$ referentes a iterações do processo evolutivo.

Observa-se que os mapas de bordas utilizados nos demais experimentos para particionar o espaço de busca apenas diminuem o custo computacional, mas não necessariamente aumentam a fração de *inliers* referente à maior superfície em cada espaço de busca. Assim, mesmo com espaços de busca com tamanhos menores que o utilizado no experimento acima (toda a imagem), o limiar $t_I = 430$ pode ser considerado adequado.

Os experimentos restantes consistiram então na determinação de valores corretos para os limiares σ , θ e C , também utilizados para calcular σ_c e θ_c . Nesta tarefa foram utilizadas as 10 imagens de treinamento da base Perceptron, e os resultados visuais das segmentações foram utilizados no refinamento dos limiares. Também foram utilizados, como realimentação (*feedback*), os resultados intermediários obtidos com a ferramenta de avaliação quantitativa, apresentada na próxima seção, aplicada a outro conjunto de imagens Perceptron contendo 30 imagens de testes.

Foi observado que valores muito pequenos para os limiares σ e θ ocasionam imprecisões na aproximação das superfícies pelo fato de poucos pontos serem utilizados. Quando valores muito grandes são estabelecidos, a aproximação também é prejudicada, mas por pontos de outras superfícies, próximos às bordas da superfície aproximada. Valores pequenos para a constante de multiplicação C fazem com que os limiares σ_c e θ_c sejam pequenos e assim o crescimento não anexa regiões ruidosas da superfície, cuja região não é totalmente extraída, o que pode causar sobre-segmentação. Por outro lado, se a constante C e, conseqüentemente, os limiares σ_c e θ_c forem muito grandes, uma região em crescimento pode anexar uma quantidade excessiva de pontos de outras regiões vizinhas. Nesse caso, as regiões resultantes da atualização do espaço de busca podem acabar sendo descartadas por apresentarem tamanhos muito pequenos, causando sub-segmentação.

Ao término desses experimentos, os valores estabelecidos para os limiares foram:

$$\sigma = 1.4 \quad \theta = 14 \quad C = 3 \quad (4.27)$$

As figuras 4.20–4.30, a seguir, mostram alguns dos resultados finais, na forma de segmentações por regiões, obtidos com o método desenvolvido e utilizando o mesmo conjunto de parâmetros, acima, para todas as imagens. Também são mostrados os resultados dos métodos USF, WSU, UB e UE, aos quais o método desenvolvido é comparado com base na avaliação quantitativa apresentada na próxima seção.

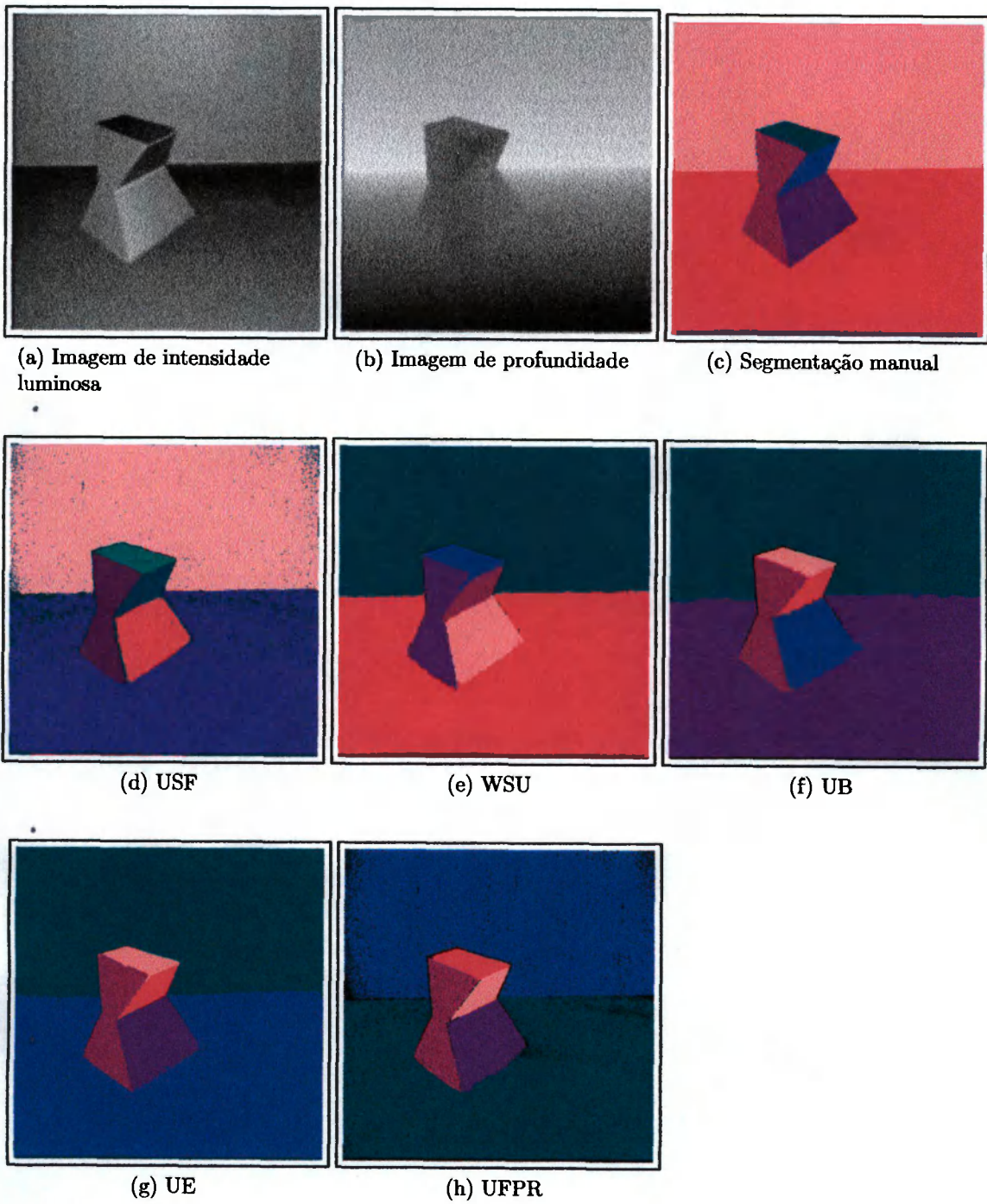


Figura 4.20: *Regiões segmentadas para imagem de testes Perceptron número 0.*

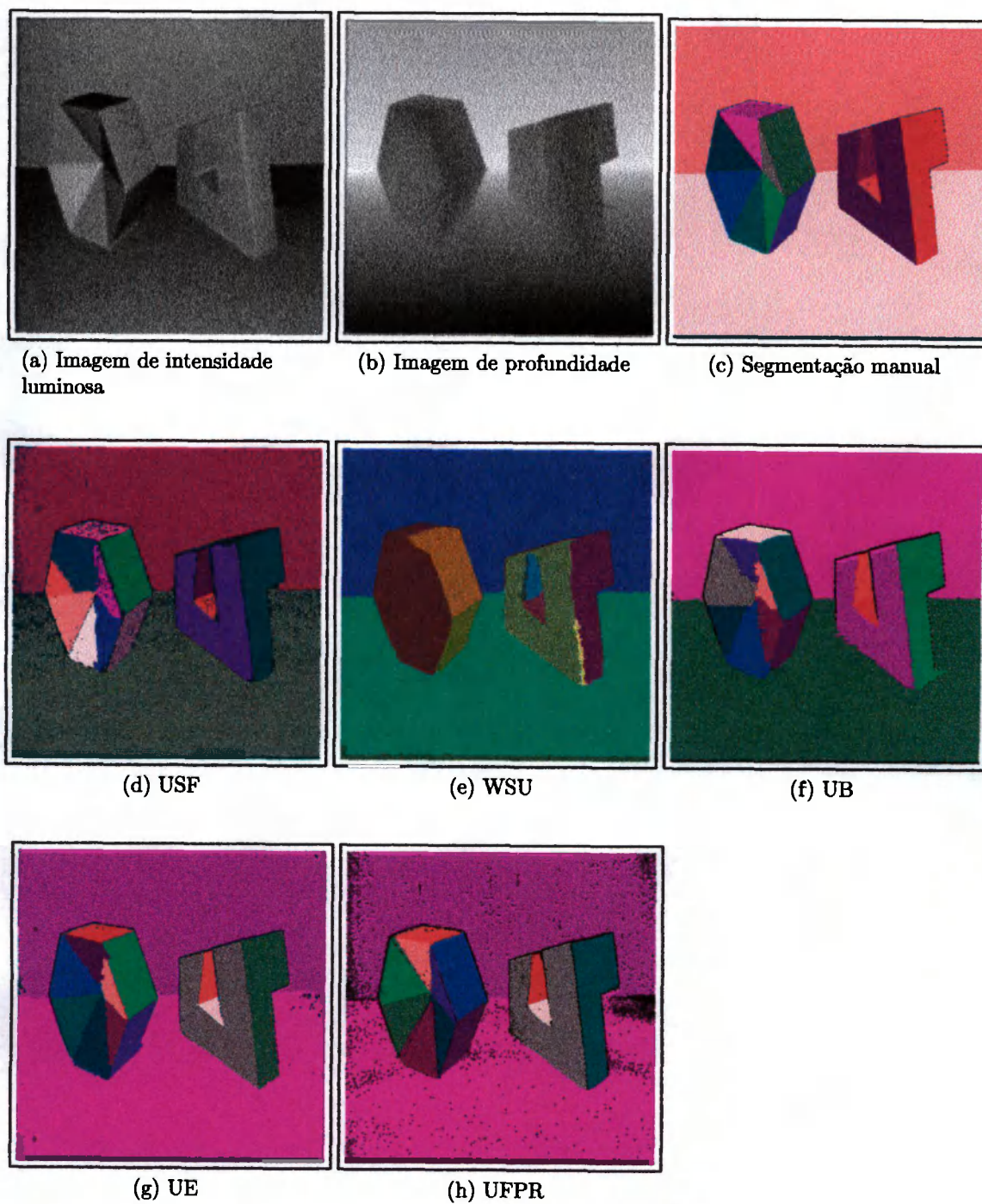


Figura 4.21: *Regiões segmentadas para imagem de testes Perceptron número 18.*

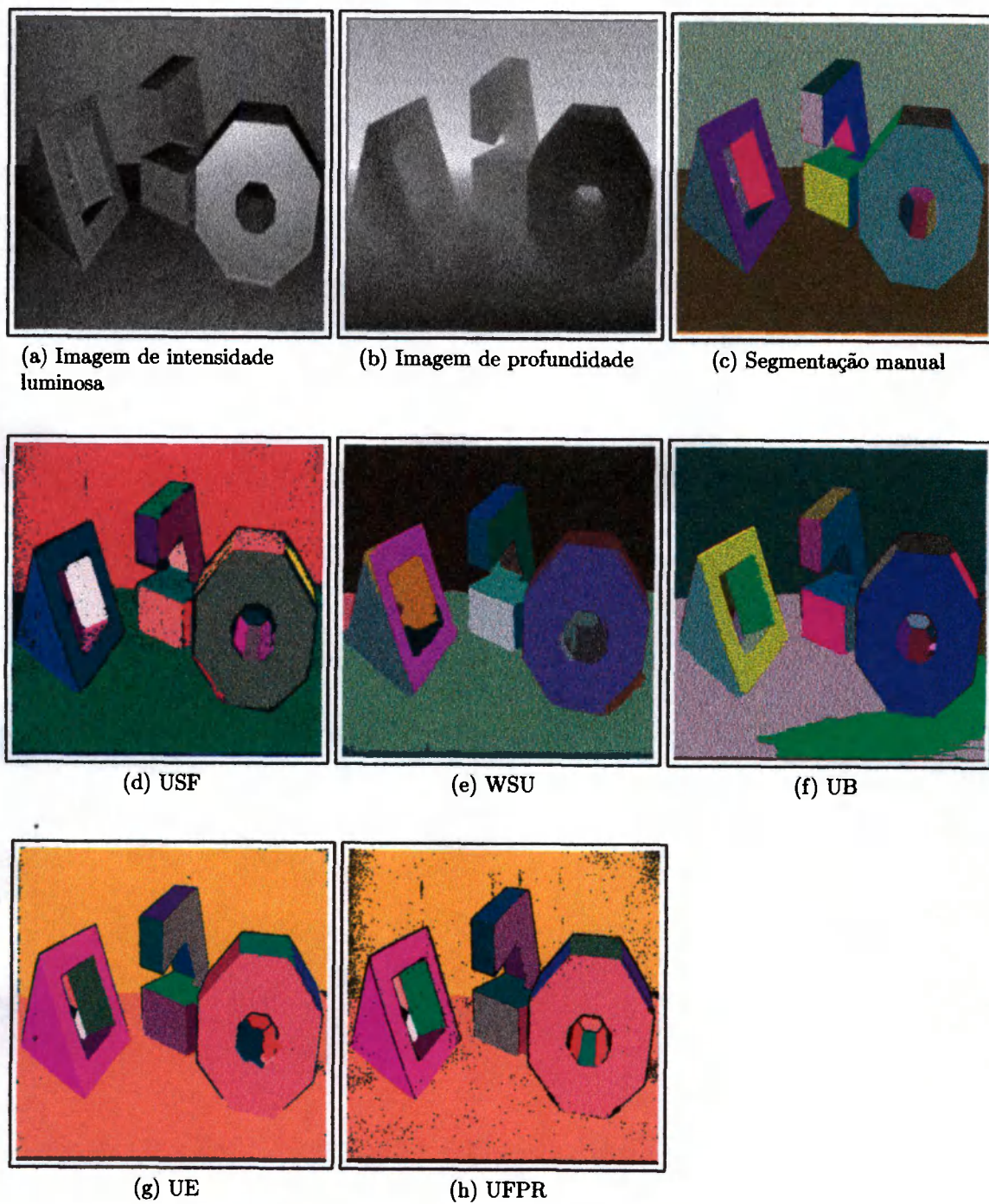


Figura 4.22: *Regiões segmentadas para imagem de testes Perceptron número 19.*

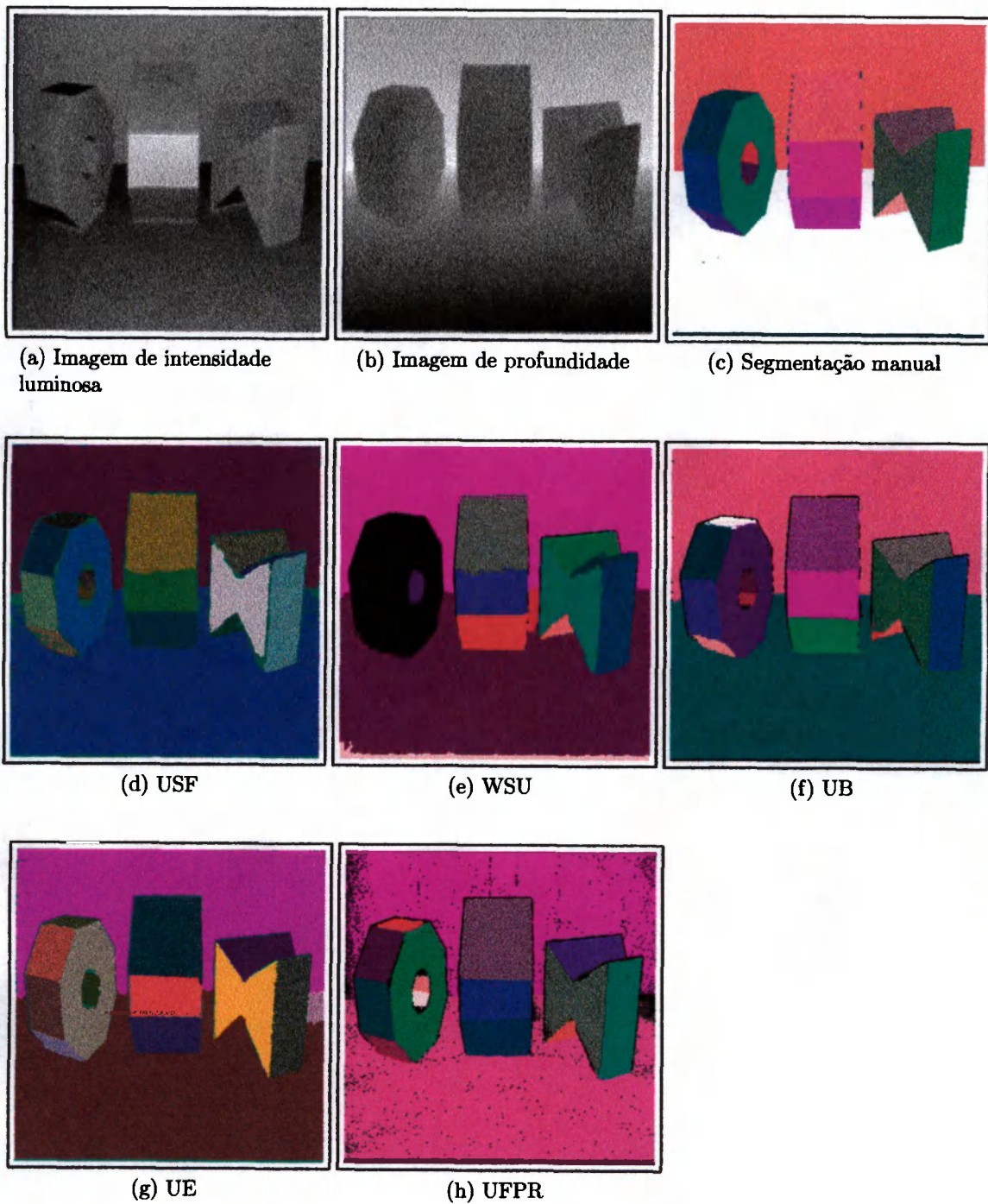


Figura 4.23: *Regiões segmentadas para imagem de testes Perceptron número 20.*

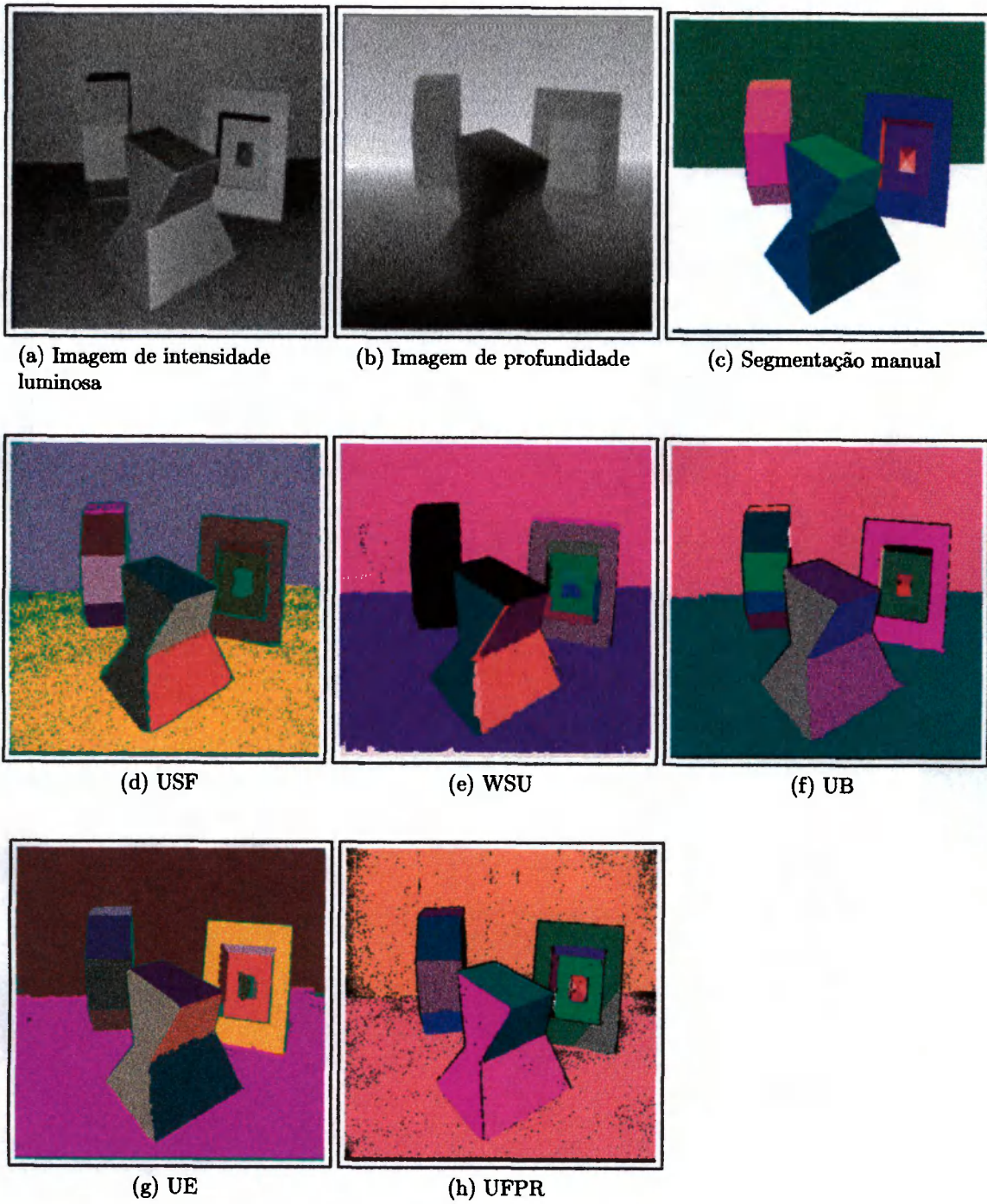


Figura 4.24: Regiões segmentadas para imagem de testes Perceptron número 22.

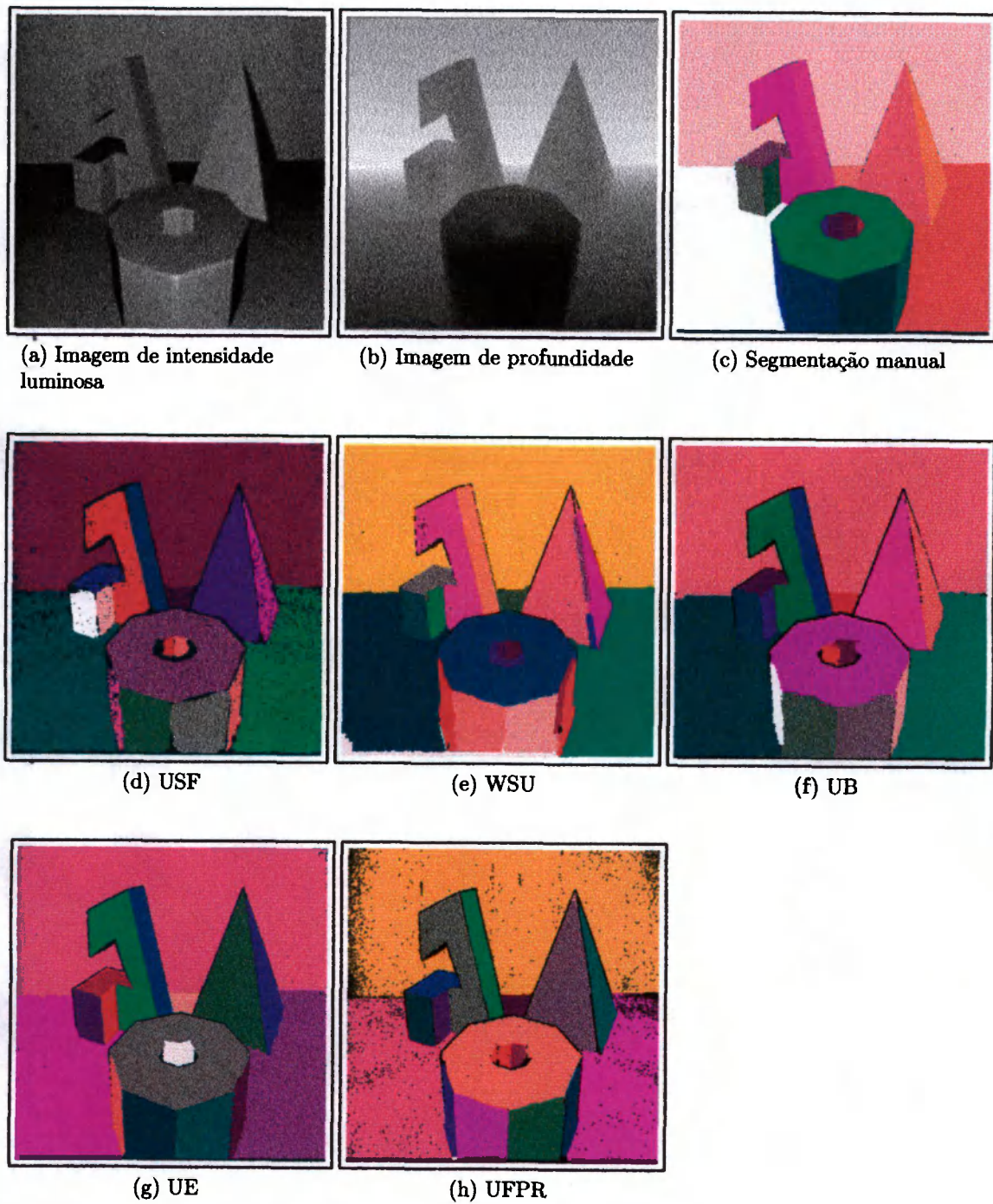


Figura 4.25: *Regiões segmentadas para imagem de testes Perceptron número 23.*

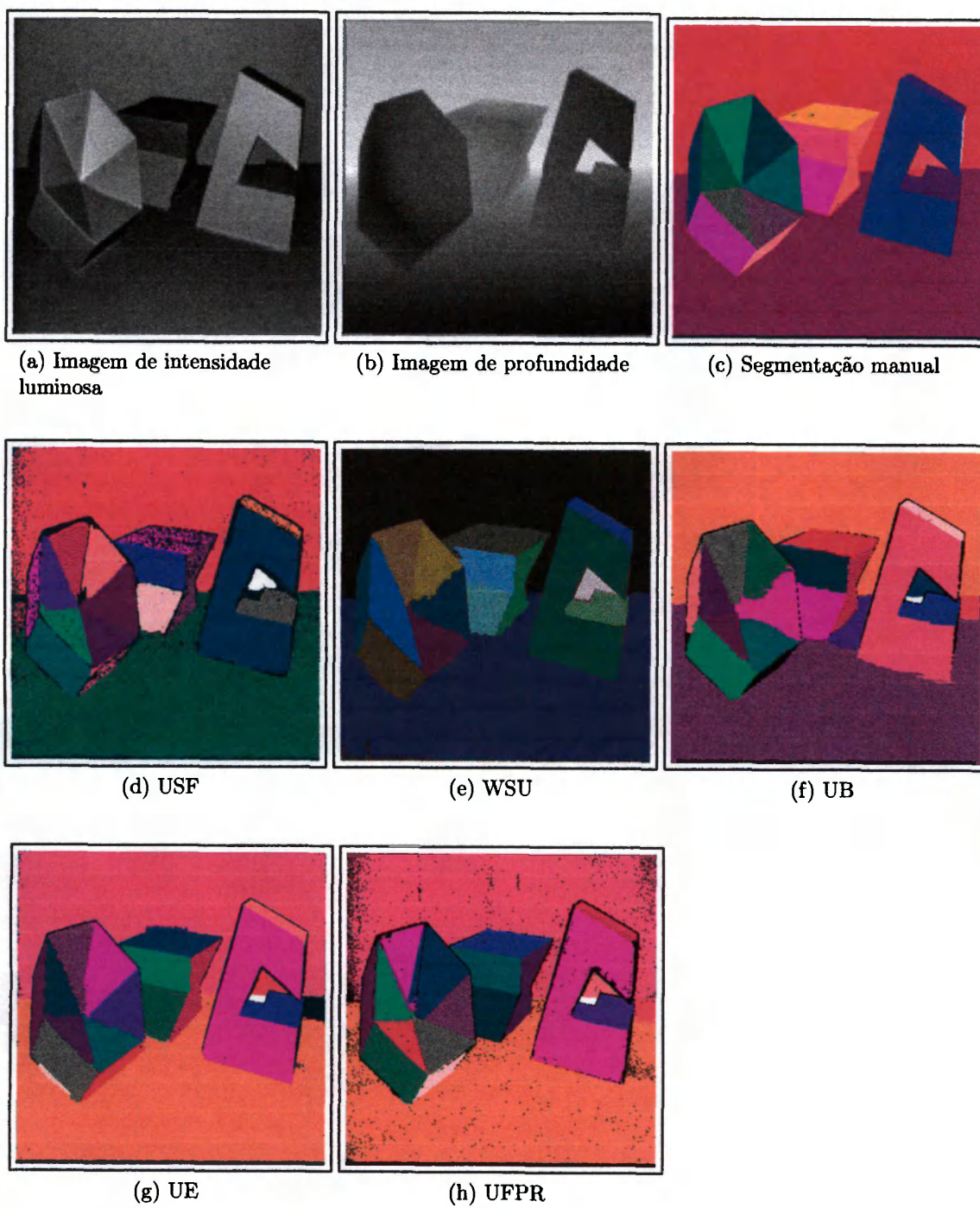


Figura 4.26: *Regiões segmentadas para imagem de testes Perceptron número 25.*



Figura 4.27: Regiões segmentadas para imagem de testes Perceptron número 26.

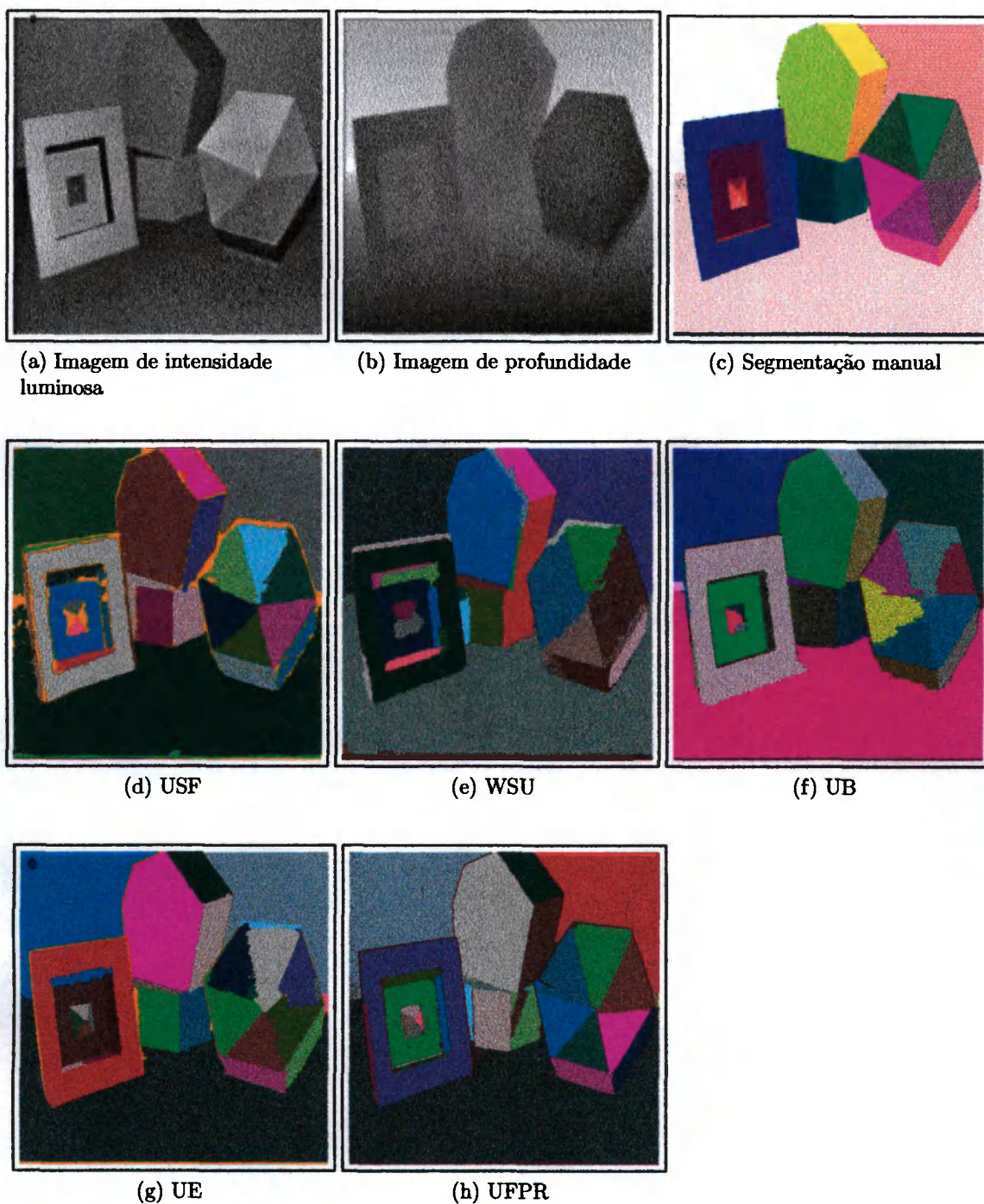


Figura 4.28: *Regiões segmentadas para imagem de testes Perceptron número 27.*

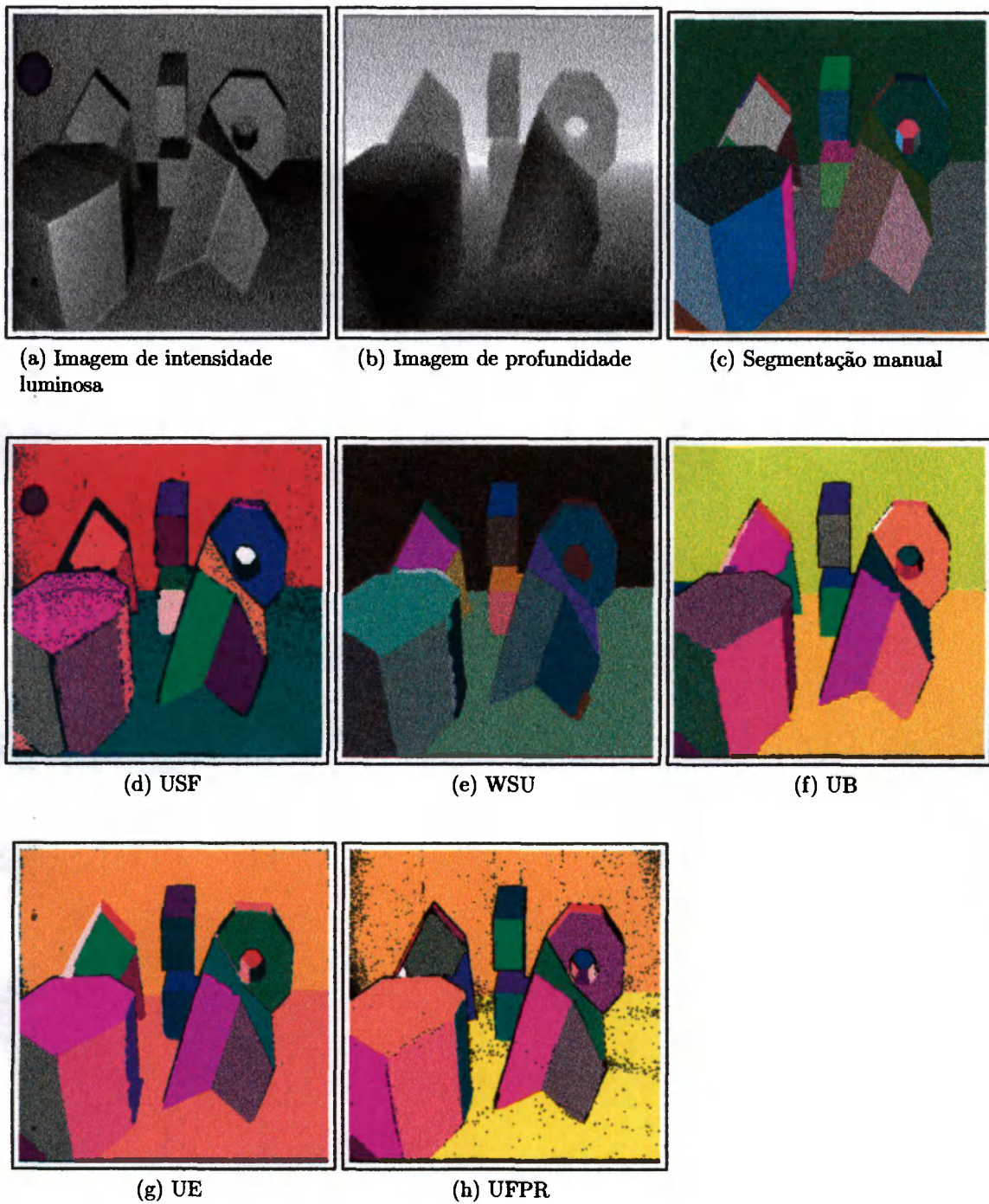


Figura 4.29: *Regiões segmentadas para imagem de testes Perceptron número 28.*

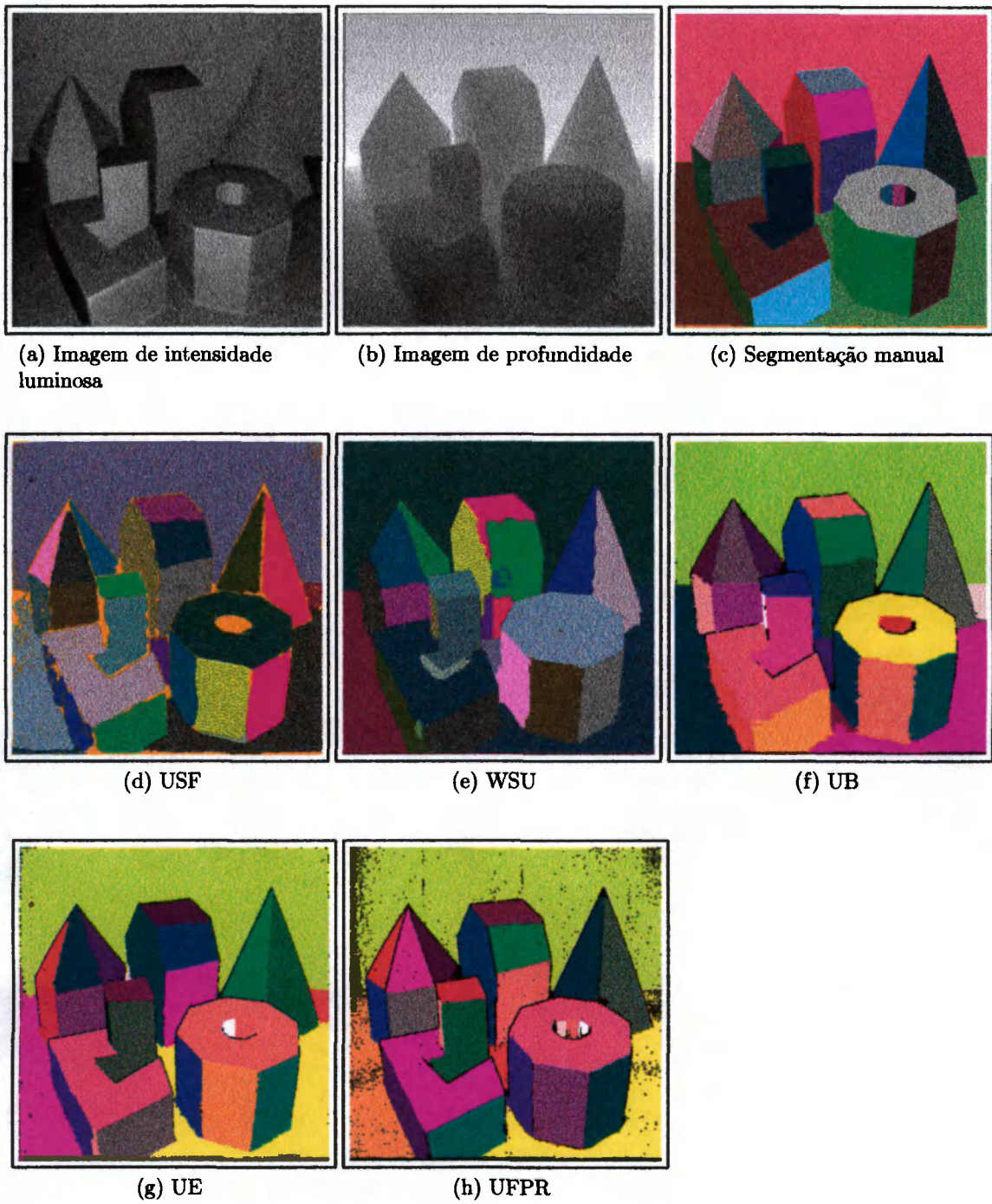


Figura 4.30: *Regiões segmentadas para imagem de testes Perceptron número 29.*

Uma análise visual dos resultados obtidos com os métodos USF, WSU, UB e UE foi realizada na seção 3.3.3. Assim, o restante desta seção trata apenas dos resultados obtidos com o método desenvolvido.

Para algumas imagens, foram obtidos melhores resultados quando da utilização de um conjunto específico de parâmetros. Mas com base nos resultados obtidos com valores fixos para os parâmetros, pode-se fazer as seguintes considerações gerais:

As regiões extraídas apresentam um grau de precisão elevado, preservando as localizações de bordas e as formas dos objetos em cena. Pode-se observar que a ocorrência de sobre-segmentação e sub-segmentação é consideravelmente pequena. Além disso, grande parte das regiões pequenas das imagens estão presentes nos resultados obtidos e com uma frequência maior que a apresentada nos resultados dos outros métodos. Também foram obtidos bons resultados na segmentação de faces apresentando poucas diferenças de orientação, como as do objeto que aparece à esquerda nas imagens de teste 18 e 25 e à direita na imagem de teste 27.

As segmentações produzidas pelo método desenvolvido também preservam melhor a topologia dos objetos em cena. Este fato é muito importante para a obtenção de uma correta representação destes objetos, por exemplo, em aplicações reconhecimento baseado em modelos CAD.

O critério do plano de decisão, também utilizado pelo algoritmo UE, colaborou para identificar precisamente as formas de algumas bordas. No entanto, algumas outras bordas apresentam formas menos precisas pelo fato de não ter havido sobreposição das regiões por ela separadas. Nestes casos, os quais ocorrem principalmente em bordas de orientação apresentando grandes descontinuidades na orientação dos vetores normais calculados localmente, o critério do plano de decisão não é aplicado pelo algoritmo.

Nos resultados obtidos também se pode observar que muitos pontos em regiões estreitas – como, por exemplo, aqueles pontos próximos às quinas (vértices) dos objetos – não foram associados a nenhuma das regiões vizinhas, permanecendo sem rótulo de superfície. Isso é causado pela imprecisão dos coeficientes normais quando calculados localmente nessas regiões. Assim, esses pontos acabam não sendo identificados como pontos compatíveis para nenhuma das regiões ao seu redor. Esse fato também contribui para a não segmentação de algumas regiões pequenas, com larguras muito inferiores às dimensões da máscara utilizada no cálculo dos vetores normais.

As regiões segmentadas apresentam também algumas regiões internas, com tamanho de apenas um *pixel*, correspondentes, em sua maioria, a pontos afetados por ruído impulsivo. Estes pontos podem ser reclassificados em um pós-processamento atribuindo-lhes, por exemplo, o mesmo rótulo que o da maioria de seus vizinhos.

Durante os experimentos, todos os estágios do algoritmo foram executados sequencialmente em um computador PC com processador Pentium IV 2 GHz e 512 MB de memória, rodando o sistema operacional Linux 2.4.18. O tempo médio para a segmentação de cada imagem de profundidade foi de aproximadamente 21 segundos, sendo que o pré-

processamento correspondeu, em média, a 4 segundos e a extração de superfícies a 17 segundos.

No entanto, caso sejam utilizadas arquiteturas paralelas, tempos de processamento menores podem ser obtidos tanto no pré-processamento como na extração de superfícies. Nesta última, uma grande redução de tempo pode ser obtida explorando o paralelismo do processo de otimização pelo AG. Estas modificações podem ser utilizadas para viabilizar a aplicação do método em tarefas de tempo real, as quais necessitam de um tempo de resposta muito baixo, como por exemplo navegação autônoma. A figura 4.31 mostra que no caso de cenas como as das imagens Perceptron, informações sobre as regiões dos dois maiores planos da imagem, juntamente com as bordas de profundidade, são na maioria das vezes suficientes para identificar as regiões correspondentes a cada objeto. Nestes exemplos, o tempo de extração de cada uma das duas regiões foi de aproximadamente 5 segundos, mas acredita-se que este tempo pode ser reduzido para menos de 1 segundo pela utilização de processamento paralelo.

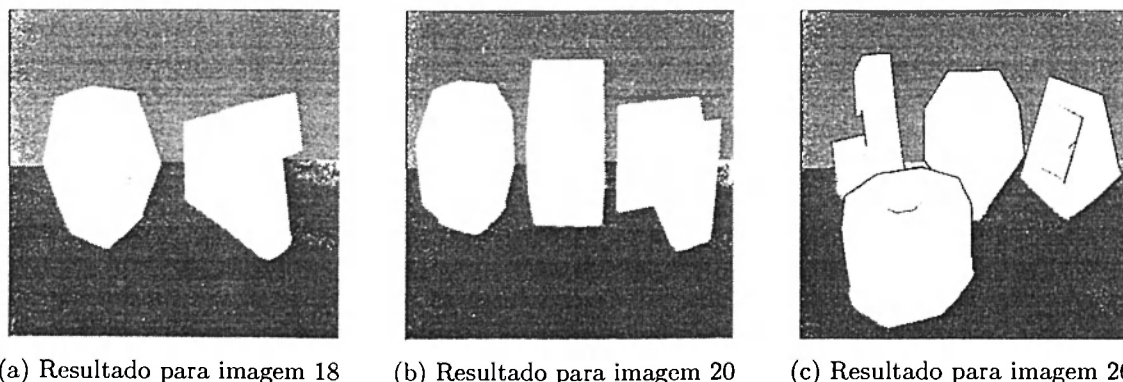


Figura 4.31: Identificação de regiões de imagens Perceptron correspondentes a objetos em cena a partir dos dois maiores planos da imagem e de bordas de profundidade.

Uma outra consideração a ser feita diz respeito ao modelo de superfície utilizado pelo método desenvolvido. Apesar de ter sido utilizado um modelo plano e de os experimentos apresentados até aqui terem envolvido apenas cenas com objetos poliédricos, o método não está restrito a essas aplicações. Com base na descrição de seus estágios de processamento, observar-se que o método pode ser estendido para utilizar outros modelos de superfície como, por exemplo, quádricas e outras superfícies curvas polinomiais. Ainda, o método pode ser aplicado na segmentação de objetos de superfícies curvas por aproximações planares, conforme exemplificado na figura 4.32. Informações sobre regiões vizinhas podem ser então utilizadas – por exemplo, utilizando modelos curvos em um estágio de pós-processamento simplificado – para a recuperação de informações sobre a verdadeira geometria dos objetos em cena.



(a) Imagem de profundidade



(b) Segmentação por planos

Figura 4.32: *Segmentação de superfícies curvas por aproximações planares.*

4.5.4 Avaliação Quantitativa

Com base nos resultados obtidos na segmentação das 30 imagens de teste Perceptron, algumas delas mostradas na seção anterior, foi realizada uma avaliação quantitativa do método desenvolvido utilizando as métricas e a ferramenta de avaliação de Hoover *et al.* (seção 2.5). Os resultados desta avaliação foram analisados comparativamente aos resultados obtidos pelos métodos UE, UB, USF e WSU. Esta análise é apresentada a seguir.

Primeiramente, para um método de segmentação ser considerado perfeito, os resultados obtidos devem apresentar uma classificação de *correta detecção* para todas as regiões segmentadas e a uma tolerância de 100%. Nenhum dos quatro métodos analisados no trabalho de Hoover *et al.* apresentou um desempenho consideravelmente próximo da perfeição, sendo que o mesmo foi constatado na análise do método desenvolvido. Isso mostra que até mesmo a segmentação de imagens contendo apenas superfícies planas, um problema aparentemente simples, ainda pode ser bastante aperfeiçoada. O grau de adequação dos resultados obtidos varia de acordo com o tipo de aplicação desejada. Para algumas aplicações, os resultados apresentados a seguir podem ser satisfatórios, enquanto para outras ainda podem precisar ser melhorados.

A uma tolerância moderada, de 80%, a melhor taxa de detecção correta obtida entre os métodos UE, UB, USF e WSU, e para as imagens de teste Perceptron, foi de 68%, apresentada pelo método UE. Na avaliação do método desenvolvido, os resultados obtidos foram comparados positivamente em relação a esses quatro outros métodos, apresentando uma taxa de detecção correta de 75%. Os valores médios para cada uma das métricas obtidos, à mesma tolerância de 80%, com estes diferentes métodos de segmentação são mostrados da tabela 4.2. Nesta tabela, tais valores são apresentados em unidades de mapeamentos entre regiões do *ground truth* e das segmentações obtidas com os métodos computacionais.

Como mostrado nos gráficos das figuras 4.33–4.37, se forem considerados todos os valores de tolerância (51%–95%) empregados pela ferramenta de avaliação durante os

experimentos, nenhum dos métodos foi sempre superior a todos os outros, no sentido de sempre apresentar, simultaneamente, os maiores valores de detecção correta e os menores valores para as métricas de erro.

O gráfico da figura 4.33 mostra que nenhum dos métodos apresentou um bom desempenho, mais que 50% de acerto, com valores de tolerância maiores que 90%. Para valores de tolerância muito próximos de 100%, a performance dos métodos decaiu rapidamente, tornando as métricas menos significativas. Esse gráfico mostra ainda que o método desenvolvido obteve uma melhor classificação de regiões corretas não só a uma tolerância de 80% mas também para todos os valores de tolerância utilizados na avaliação.

grupo de pesquisa	regiões no <i>ground truth</i>	deteção correta	dif. angular (desv. padrão)	sobre-segmentação	sub-segmentação	perdida	ruído
USF	14,6	8,9	2,7° (1,8)	0,4	0,0	5,3	3,6
WSU	14,6	5,9	3,3° (1,6)	0,5	0,6	6,7	4,8
UB	14,6	9,6	3,1° (1,7)	0,6	0,1	4,2	2,8
UE	14,6	10,0	2,6° (1,5)	0,2	0,3	3,8	2,1
UFPR	14,6	11,0	2,5° (1,7)	0,3	0,1	3,0	2,5

Tabela 4.2: Avaliação média para os métodos de segmentação aplicados às imagens de testes Perceptron à tolerância de 80%.

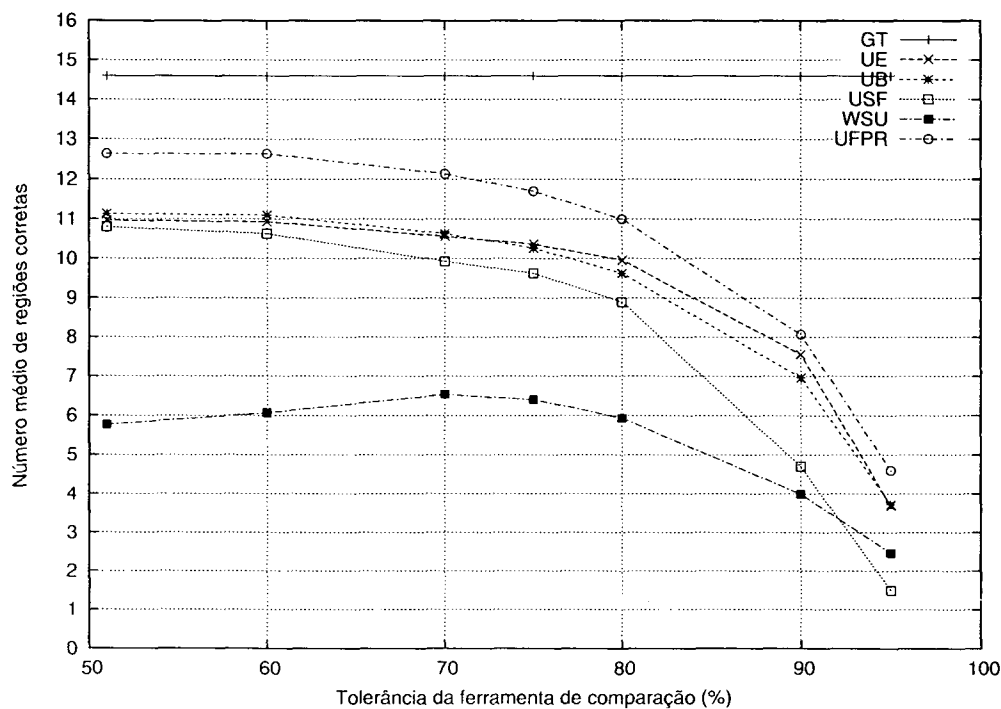


Figura 4.33: Número médio de regiões corretas de acordo com nível de tolerância.

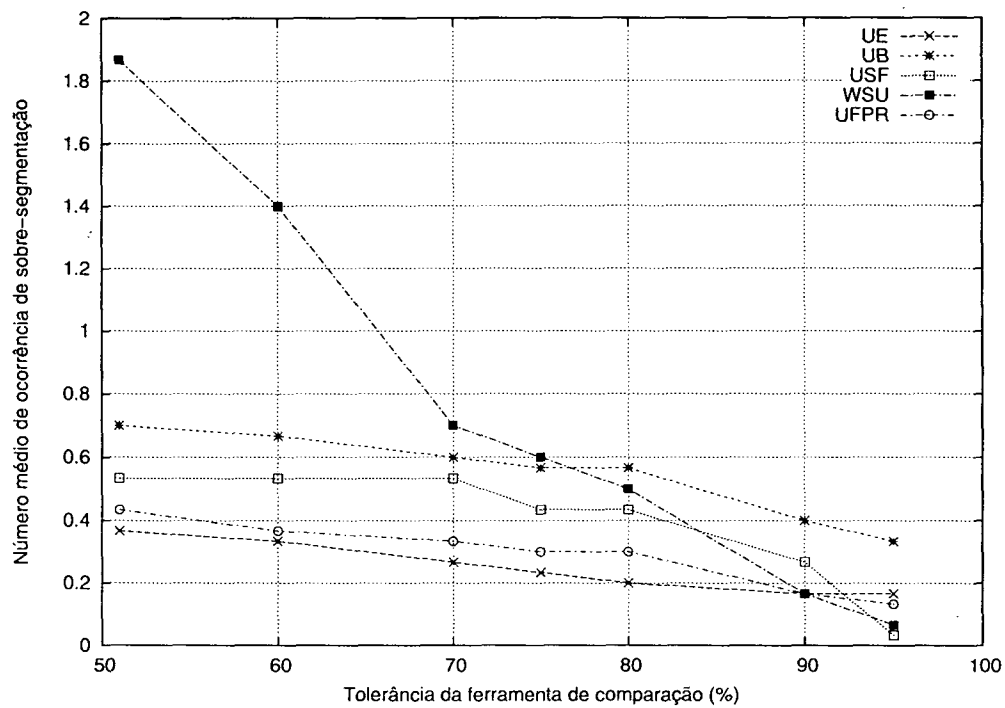


Figura 4.34: Ocorrência de sobre-segmentação em função do nível de tolerância.

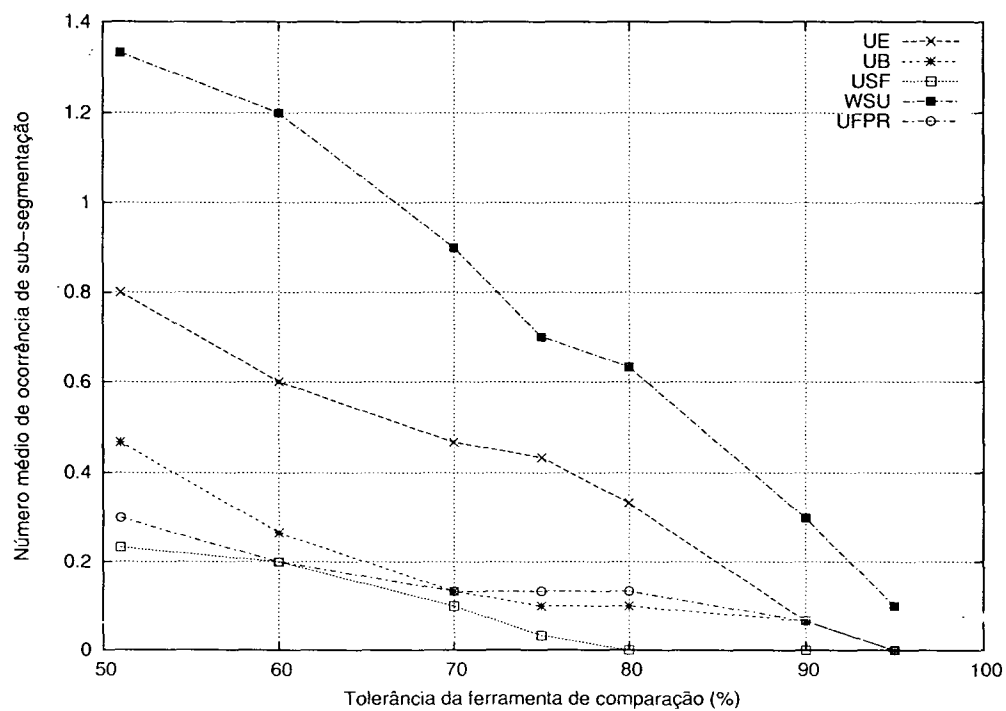


Figura 4.35: Ocorrência de sub-segmentação em função do nível de tolerância.

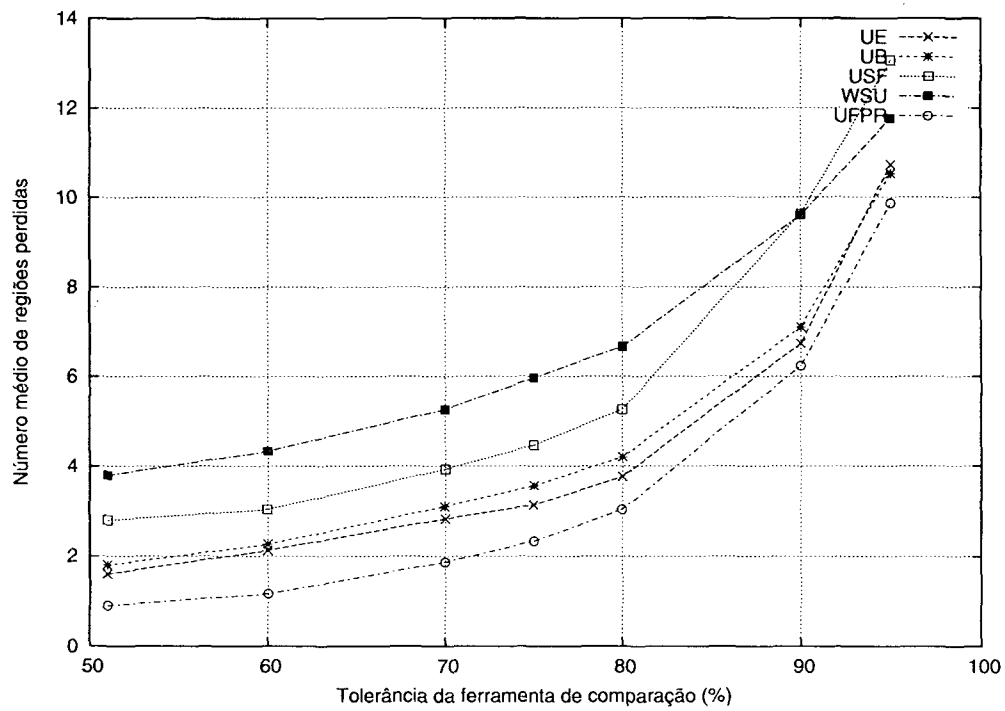


Figura 4.36: Número médio de regiões perdidas de acordo com nível de tolerância.

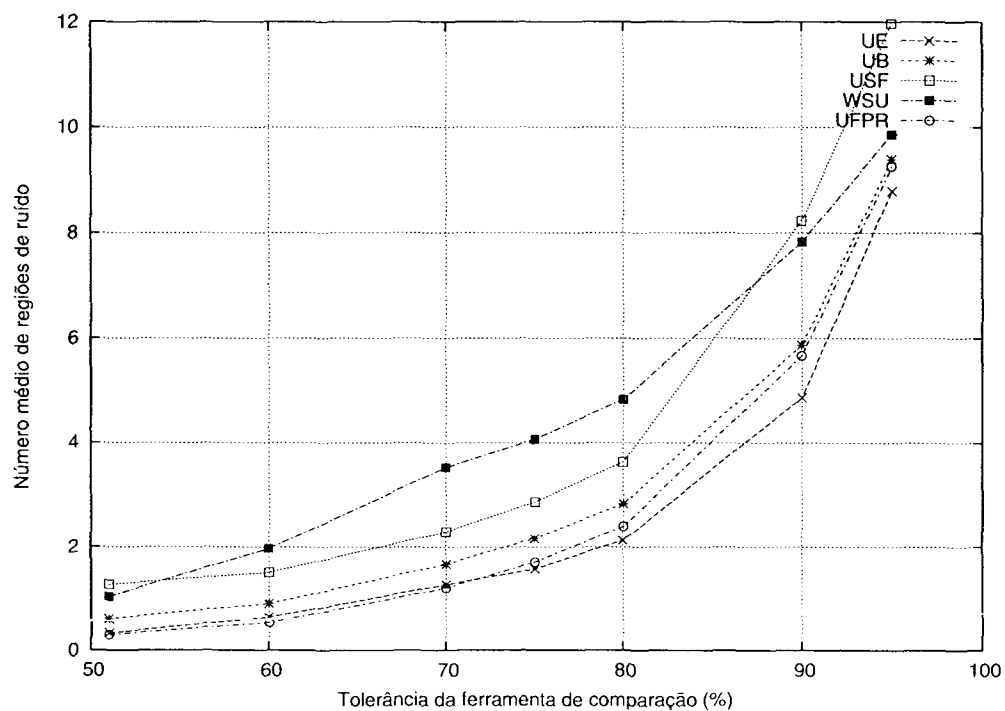


Figura 4.37: Número médio de regiões de ruído de acordo com nível de tolerância.

Como mostrado na tabela 4.2, o método desenvolvido apresentou uma menor diferença angular entre as orientações de regiões vizinhas. Esse melhor desempenho na recuperação de informação geométrica sobre os objetos em cena pode ser atribuído ao método de estimativa robusto desenvolvido. Este fato é atrativo principalmente para aplicações como engenharia reversa de modelos onde a precisão das informações geométricas obtidas é de grande importância.

Algumas observações feitas no trabalho de Hoover *et al.*, e que podem ser constatadas na tabela 4.2, também são aplicáveis ao método desenvolvido: (1) os valores obtidos nos casos de regiões perdidas e de ruídos são maiores que os valores de ocorrência de sobre- e sub-segmentação; (2) a ocorrência de sobre-segmentação tem maior predominância que a ocorrência de sub-segmentação, enquanto casos de regiões perdidas são mais frequentes que casos de regiões de ruído.

Nos gráficos das figuras 4.34 e 4.35, pode ser observado que a classificação de ocorrências de sobre- e sub-segmentação diminui conforme aumenta o valor de tolerância. Isso ocorre porque aumenta o número de regiões classificadas como perdidas ou como ruído, conforme mostrado nos gráficos das figuras 4.36 e 4.37.

Entre as considerações finais feitas no trabalho de Hoover *et al.*, ressaltou-se que o fato de as performances dos métodos serem insatisfatórias a valores de tolerância maiores que 90%, sugere ser necessária uma melhor precisão na localização das bordas das regiões. Isso também se aplica ao método desenvolvido. Além disso, foi observado também que muitas regiões pequenas das imagens, com tamanhos menores que 1000 *pixels*, eram classificadas como perdidas pelos métodos de segmentação.

Quanto à última consideração, um avanço obtido com o método desenvolvido é a melhor segmentação dessas regiões pequenas das imagens, o que proporcionou ao método menores valores para a métrica de regiões perdidas, como mostrado na tabela 4.2 e no gráfico da figura 4.36. Este fato também contribuiu para a obtenção de melhores valores de detecção correta de regiões. No entanto, ainda existe espaço para melhorias.

4.6 Conclusão

Neste capítulo foram apresentados os trabalhos de pesquisa realizados sobre segmentação de imagens de profundidade por extração de superfícies utilizando métodos de aproximação robustos. Esses trabalhos de investigação resultaram no desenvolvimento de um novo método de segmentação, o qual foi direcionado de modo a resolver alguns problemas apresentados por outros métodos, como: (1) a interferência de *pseudo-inliers* no processo de extração; (2) a convergência precoce na aceleração do processo de otimização de hipóteses; (3) a incorreta localização de bordas; e (4) a não segmentação de regiões pequenas das imagens.

Assim, uma das contribuições apresentadas foi uma nova versão do método robusto MSAC, a qual agrega informação de orientação de superfície, calculada localmente, para

possibilitar a classificação de *pseudo-inliers* como *outliers*. Este fato contribui para evitar a ocorrência de sobre-segmentação e melhor definir as formas das regiões extraídas.

Outra contribuição consiste em um novo conjunto de parâmetros – incluindo uma nova combinação de tipos das operações de seleção, *crossover* e mutação – para AGs empregados na aceleração do processo de busca realizado previamente à extração. Durante os experimentos realizados, o AG utilizando esses parâmetros foi capaz de melhor acelerar o processo de otimização e apresentou um índice de convergência precoce menor que o apresentado por um AG com o conjunto de parâmetros frequentemente utilizados.

O novo método de segmentação de imagens de profundidade por extração de superfícies combina técnicas de detecção de bordas e de crescimento de regiões. Durante a extração, o método robusto e o AG desenvolvidos são aplicados na identificação de uma região semente e dos parâmetros da superfície a ser extraída. O processo de crescimento da região semente é reversível e o critério do plano de decisão utilizado possibilitou uma boa precisão quanto à localização das bordas das superfícies extraídas.

Este método de segmentação foi comparado positivamente em relação a outros quatro algoritmos bastante conhecidos da literatura. Como mostrado nos experimentos e na avaliação quantitativa realizados, o método desenvolvido também foi capaz de melhor segmentar as regiões pequenas das imagens, as quais são frequentemente perdidas pela maioria dos outros métodos.

As contribuições apresentadas constituem avanços relevantes na segmentação de imagens de profundidade. Ressalta-se que o método não está limitado ao uso do modelo planar ou apenas à segmentação de objetos polidricos. O método pode ser estendido para utilizar modelos curvos, ou ainda, ser utilizado para segmentar objetos de superfícies curvas através de aproximações planares.

Alguns problemas identificados estão relacionados, principalmente, ao cálculo local dos vetores normais. Em algumas regiões com dimensões menores que as dimensões da máscara utilizada nesse cálculo, os coeficientes estimados para os vetores apresentaram pouca precisão, o que prejudicou a segmentação de algumas regiões pequenas ou de partes de regiões maiores. A localização de algumas bordas de regiões maiores também foi prejudicada por imprecisões no cálculo dos vetores normais.

Como trabalhos futuros, além do tratamento dos problemas já identificados, pode-se citar, principalmente, a utilização de outros tipos de modelos para a segmentação de superfícies curvas. O maior número de graus de liberdade destas superfícies requer um maior número de genes nos cromossomos do AG, o que pode implicar na necessidade de novas melhorias de performance. Investigações também podem ser feitas para se calcular automaticamente os valores de limiar utilizados na extração – por exemplo, tendo como base informações estatísticas da imagem e os limiares de bordas automáticos – e tornar o método mais independente de supervisão humana. Ainda, estudos sobre formas de integração de dados de intensidade luminosa na extração de regiões podem ser realizados com o objetivo de melhorar os resultados obtidos.

Capítulo 5

Conclusão e Trabalhos Futuros

Sistemas de visão computacional buscam reproduzir as capacidades do sistema de visão humano com o objetivo de reconhecer objetos e interpretar uma cena descrita por uma imagem digital. Apesar de vários sensores e diferentes métodos computacionais terem sido reportados na literatura desta área de pesquisa, os sistemas de visão computacional atualmente desenvolvidos ainda enfrentam limitações que inviabilizam certos tipos de aplicação. Muitos destes sistemas são utilizados para tarefas bastante específicas e não apresentam um desempenho ótimo ou flexibilidade suficiente para serem empregados em diferentes situações sem necessitar de esforços de desenvolvimento adicionais.

Em parte, isso é devido a problemas encontrados durante a segmentação de imagens, um processo fundamental para que o reconhecimento dos objetos em cena seja realizado de forma confiável. A segmentação de imagens ainda constitui um dos maiores desafios na área de pesquisa em visão computacional porque deve identificar, com precisão, as formas e a localização das bordas dos objetos em cena. Para isso, este processo depende fortemente da correta extração de características dos objetos a partir de dados de entrada apresentando degradações, denominadas ruídos. Neste contexto, imagens de profundidade vêm sendo cada vez mais utilizadas por apresentarem informações 3D explícitas, as quais são úteis para a caracterização geométrica dos dados sobre as cenas observadas.

O principal objetivo do trabalho aqui apresentado é contribuir para a solução do problema de segmentação de imagens de profundidade, já que este processo ainda apresenta deficiências mesmo para cenas, aparentemente simples, contendo apenas superfícies planas.

Na busca por soluções para este problema, duas abordagens foram investigadas: a primeira concentrou-se na segmentação por detecção de bordas baseada nas curvaturas de superfície H e K e integrando dados de intensidade luminosa correspondentes às mesmas cenas nas imagens de profundidade; a segunda abordagem utilizou um método de estimativa robusto e um algoritmo genético para a segmentação da imagem de profundidade através da extração de regiões correspondentes às superfícies planas das cenas observadas.

Estes trabalhos também fazem parte de uma projeto mais amplo, em desenvolvimento.

pelo grupo de pesquisa IMAGO, e os resultados obtidos vêm sendo utilizados como suporte a este projeto na extração de características e segmentação de imagens.

As seguintes contribuições são apresentadas como resultado dos estudos realizados:

- Indicativo do impacto positivo que a integração de dados de intensidade luminosa exerce na segmentação da imagem de profundidade correspondente à mesma cena.
- Um método inédito de detecção de bordas a partir de mapas de sinais das curvaturas H e K , os quais haviam sido anteriormente utilizados apenas na identificação de regiões, mas não bordas, conforme apresentado neste trabalho. Também foi apresentado um método de detecção de bordas utilizando somente os valores da curvatura H também calculados a partir de ambos os tipos de imagens. Os resultados experimentais mostram que as bordas detectadas com os dois métodos melhor preservam as formas dos objetos em cena, quando comparadas as formas das regiões obtidas com quatro outros métodos de segmentação apresentados na literatura.
- Um novo método de estimativa robusto agregando informação de orientação de superfície para aplicação em segmentação de imagens de profundidade de modo a evitar alguns casos de incorreto particionamento de superfícies, melhorando as formas das regiões extraídas.
- Um conjunto de parâmetros de algoritmos genéticos para aplicação, juntamente com métodos de estimativa robusta, na extração de superfícies em imagens de profundidades. Tais parâmetros apresentaram um melhor desempenho na aceleração do processo de otimização e na preservação da diversidade genética da população de cromossomos, o que diminui a possibilidade de convergência precoce.
- Um método de segmentação de imagens de profundidade por extração de superfícies, utilizando o método de estimativa robusto desenvolvido e um algoritmo genético com o novo conjunto de parâmetros estabelecido. Através de uma avaliação quantitativa utilizando um conjunto de métricas e uma base de imagens em comum, os resultados obtidos na segmentação foram comparados positivamente aos resultados de outros quatro métodos de segmentação muito conhecidos da literatura, sendo que o método desenvolvido foi capaz de melhor segmentar regiões pequenas das imagens.

As contribuições apresentadas constituem avanços relevantes para a solução do problema da segmentação de imagens de profundidades, principalmente quanto à melhor preservação da topologia dos objetos em cena, o que por sua vez auxilia processos posteriores de representação e reconhecimento desses objetos. No entanto, como os resultados apresentados na avaliação quantitativa sugerem, ainda existe espaço para novas melhorias, até mesmo para cenas contendo apenas objetos poliédricos. Essas melhorias devem proporcionar, principalmente, localizações de bordas mais precisas aos resultados de segmentação.

Além dos trabalhos futuros discutidos no final dos dois últimos capítulos, referentes a melhoramentos para os métodos desenvolvidos, deve ser enfatizado o trabalho mais importante que é a integração dos métodos de detecção de bordas baseados nas curvaturas H e K com o método de extração de superfícies. Ou seja, a utilização dos mapas de bordas produzidos por um dos dois métodos baseados em curvaturas para proporcionar uma melhor definição das formas das regiões extraídas pelo outro método. Além disso, os mapas de sinais HK podem ser utilizados na escolha do modelo de superfície utilizado na extração, quando imagens contendo objetos poliédricos e de superfícies curvas forem processadas.

Uma expectativa para essa integração é através da fusão de um dos métodos de detecção de bordas com o estágio de pré-processamento do método de extração de superfícies. Para isso, a princípio, ao menos um dos fatos abaixo deve ser possível:

- Obtenção de resultados mais precisos para os vetores normais calculados a partir dos valores de derivadas parciais;
- Cálculo de valores mais precisos para as curvaturas H e K a partir dos desvios angulares dos vetores normais calculados no estágio de pré-processamento.

Finalmente, outro trabalho futuro importante é a investigação sobre a utilização dos dados de intensidade luminosa na avaliação da função de custo do método robusto.

Referências Bibliográficas

- [Bellon *et al.*, 1999] Bellon, O., Direne, A., e Silva, L. (1999). Edge detection to guide range image segmentation by clustering techniques. In *Proceedings of the IEEE International Conference on Image Processing*, páginas 725–729.
- [Bellon e Silva, 2002] Bellon, O. e Silva, L. (2002). New improvements on range image segmentation by edge detection. *IEEE Signal Processing Letters*, 9(2):43–45.
- [Besl, 1988] Besl, P. (1988). *Surfaces in Range Image Understanding*. Springer-Verlag.
- [Besl, 1989] Besl, P. (1989). *Active Optical Range Imaging Sensors, Advances in Machine Vision*. Springer-Verlag.
- [Besl e Jain, 1985] Besl, P. e Jain, R. (1985). Three-dimensional object recognition. *Computing Surveys*, 17(1):75–145.
- [Besl e Jain, 1988] Besl, P. e Jain, R. (1988). Segmentation through variable-order surface fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(2):167–192.
- [Cai, 1989] Cai, L. (1989). A small leakage model for diffusion smoothing of range data. In *Proceedings of the International Joint Conference on Artificial Intelligence*, páginas 1585–1590.
- [Celinski, 1998] Celinski, T. (1998). Métodos de agrupamento: uma abordagem comparativa com aplicação em segmentação de imagens de profundidade. *Dissertação de Mestrado*, Programa de Pós-Graduação em Informática da Universidade Federal do Paraná, Curitiba, PR.
- [Celinski e Bellon, 1998] Celinski, T. e Bellon, O. (1998). Métodos de agrupamento, Tutorial. In *XI Simpósio Brasileiro em Computação Gráfica e Processamento de Imagens*, Rio de Janeiro, RJ.
- [Chen e Liu, 1999] Chen, Y. e Liu, C. (1999). Quadric surface extraction using genetic algorithms. *Computer-Aided Design*, 31(2):101–110.
- [Fan *et al.*, 1987] Fan, T., Medioni, G., e Nevatia, R. (1987). Segmented description of 3-d surfaces. *IEEE Transactions on Robotics and Automation*, 3(6):527–538.

- [Fischler e Bolles, 1981] Fischler, M. e Bolles, R. (1981). Random Sample Consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–395.
- [Fitzgibbon *et al.*, 1997] Fitzgibbon, A., Eggert, D., e Fisher, R. (1997). High-level CAD model acquisition from range images. *Computer-Aided Design*, 29(4):321–330.
- [Flynn e Jain, 1989] Flynn, P. e Jain, A. (1989). On reliable curvature estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, páginas 110–116.
- [Gonzalez e Woods, 2000] Gonzalez, R. e Woods, R. (2000). *Processamento de Imagens Digitais*. Editora Edgard Blücher.
- [Gotardo *et al.*, 2002] Gotardo, P., Bellon, O., e Silva, L. (2002). Advances on range image segmentation: Edge detection using surface curvature from intensity and range images and region extraction using a genetic algorithm-based robust estimator. In *I Workshop de Teses e Dissertações em Computação Gráfica e Processamento de Imagens*, in *XV Simpósio Brasileiro de Computação Gráfica e Processamento de Imagens*, Fortaleza, CE. A ser publicado na Revista Scientia, Editora Unisinos, RS.
- [Haddon, 1988] Haddon, J. (1988). Generalized threshold selection for edge detection. *Pattern Recognition*, 21(3):195–203.
- [Hakala *et al.*, 1981] Hakala *et al.*, D. (1981). Natural quadrics in mechanical design. *SIGGRAPH/81*.
- [Hilton *et al.*, 1995] Hilton, A., Illingworth, J., e Winderatt, T. (1995). Statistics of surface curvature estimates. *Pattern Recognition*, 28(8):1201–1221.
- [Hoffman e Jain, 1987] Hoffman, R. e Jain, A. (1987). Segmentation and classification of range images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):608–620.
- [Hoover *et al.*, 1996] Hoover, A., Jean-Baptiste, G., Jiang, X., Flynn, P., Bunke, H., Goldgof, D., Bowyer, K., Eggert, D., Fitzgibbon, A., e Fisher, R. (1996). An experimental comparison of range image segmentation algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):673–689.
- [Huber, 1981] Huber, P. (1981). *Robust Statistics*. John Wiley.
- [Jain *et al.*, 1995] Jain, R., Kasturi, R., e Schunck, B. (1995). *Machine Vision*. McGraw-Hill.
- [Jarvis, 1983] Jarvis, R. (1983). A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(2):122–139.

- [Jiang *et al.*, 2000] Jiang, X., Bowyer, K., Morioka, Y., Hiura, S., Sato, K., Inokuchi, S., Bock, M., Guerra, C., Loke, R., e du Buf, J. (2000). Some further results of experimental comparison of range image segmentation algorithms. In *Proceedings of the International Conference on Pattern Recognition*, volume 4, páginas 877–881.
- [Lawson e Hanson, 1974] Lawson, C. e Hanson, R. (1974). *Solving Least Squares Problems*. Prentice-Hall.
- [Lee *et al.*, 1998] Lee, K., P., M., e R.H., P. (1998). Robust adaptive segmentation of range images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(2):200–205.
- [Man *et al.*, 1996] Man, K., Tang, K., e Kwong, S. (1996). Genetic algorithms: Concepts and applications. *IEEE Transactions on Industrial Electronics*, 43(5):519–534.
- [Meer *et al.*, 2000] Meer, P., Stewart, C., e Tyler, D. (2000). Robust computer vision: An interdisciplinary challenge. *Computer Vision and Image Understanding*, 78(1):1–7.
- [Miller e Stewart, 1996] Miller, J. e Stewart, C. (1996). Muse: Robust surface fitting using unbiased scale estimates. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, páginas 300–306.
- [Nitzan *et al.*, 1977] Nitzan, D., Brain, A., e Duda, R. (1977). The measurement and use of registered reflectance and range data in scene analysis. *Proceedings of the IEEE*, 65(2):206–220.
- [Powell *et al.*, 1998] Powell, M., Bowyer, K., Jiang, X., e Bunke, H. (1998). Comparing curved-surface range image segmenters. In *Proceedings of the IEEE International Conference on Computer Vision*, páginas 286–291.
- [Press *et al.*, 1992] Press, W., Teukolsky, S., W.T., V., e Flannery, B. (1992). *Numerical Recipes in C - The Art of Scientific Computing*. Cambridge University Press, second edition.
- [Rosenfeld, 1969] Rosenfeld, A. (1969). Picture processing by computer. *ACM Computing Surveys*, 1(3):147–176.
- [Roth e Levine, 1993] Roth, G. e Levine, M. (1993). Extracting geometric primitives. *Computer Vision, Graphics, and Image Processing*, 58(1):1–22.
- [Roth e Levine, 1994] Roth, G. e Levine, M. (1994). Geometric primitive extraction using a genetic algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):901–905.
- [Rousseeuw, 1984] Rousseeuw, P. (1984). Least median of squares regression. *Journal of the American Statistics Association*, 79:871–880.

- [Rousseeuw e Leroy, 1987] Rousseeuw, P. e Leroy, A. (1987). *Robust Regression and Outlier Detection*. John Wiley.
- [Silva, 2000] Silva, L. (2000). Segmentação de imagens de profundidade por detecção de bordas. *Dissertação de Mestrado*, Programa de Pós-Graduação em Informática da Universidade Federal do Paraná, Curitiba, PR.
- [Silva, 2002] Silva, L. (2002). *Segmentação e Representação de Objetos em Imagens de Profundidade*. Tese de Doutorado, Curso de Pós-Graduação em Engenharia Elétrica e Informática Industrial do Centro Federal de Educação Tecnológica do Paraná. proposta.
- [Silva et al., 2001] Silva, L., Bellon, O., e Gotardo, P. (2001). Edge-based image segmentation using curvature sign maps from reflectance and range images. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, páginas 730–734, Tessalônica, Grécia.
- [Silva et al., 2002] Silva, L., Bellon, O., e Gotardo, P. (2002). A global-to-local approach for robust range image segmentation. In *Proceedings of the IEEE International Conference on Image Processing*, volume 1, páginas 773–776, Rochester, Nova Iorque.
- [Sonka et al., 1998] Sonka, M., Hlavac, V., e Boyle, R. (1998). *Image Processing, Analysis, and Machine Vision*. Brooks/Cole.
- [Stewart, 1995] Stewart, C. (1995). MINPRAN: A new robust estimator for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(10):925–938.
- [Stewart, 1999] Stewart, C. (1999). Robust parameter estimation in computer vision. *SIAM Review*, 41(3):513–537.
- [Taubin, 1991] Taubin, G. (1991). Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1115–1138.
- [Torr e Zisserman, 2000] Torr, P. e Zisserman, A. (2000). MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156.
- [Trucco, 1992] Trucco, E. (1992). On shape-preserving boundary conditions for diffusion smoothing. In *Proceedings of the IEEE International Conference on Robotics and Automation*, páginas 1690–1694.
- [Trucco e Fisher, 1992] Trucco, E. e Fisher, R. (1992). Computing surface-based representations from range images. In *Proceedings of the IEEE International Symposium on Intelligent Control*, páginas 275–280, Glasgow, Escócia.

- [Umasuthan e Wallace, 1996] Umasuthan, M. e Wallace, A. (1996). Outlier removal and discontinuity preserving smoothing of range data. *IEEE Vision, Image and Signal Processing*, 143(3):191–200.
- [Vieira, 2002] Vieira, E. (2002). Mineração de imagens: Conceito e aplicação em sistemas de recuperação de imagens por conteúdo. *Dissertação de Mestrado*, Programa de Pós-Graduação em Informática da Universidade Federal do Paraná, Curitiba, PR.
- [Vieira et al., 2002] Vieira, E., Bellon, O., e Silva, L. (2002). Mineração de imagens: conceitos, aplicações e desafios, Tutorial. In *XV Simpósio Brasileiro em Computação Gráfica e Processamento de Imagens*, Fortaleza, CE.
- [Várady et al., 1997] Várady, T., Martin, R., e Coxt, J. (1997). Reverse engineering of geometric models – An introduction. *Computer-Aided Design*, 29(4):255–268.
- [Wang e Cheng, 1992] Wang, Y. F. e Cheng, D. I. (1992). Three-dimensional shape construction and recognition by fusing intensity and structured lighting. *Pattern Recognition*, 25(12):1411–1425.
- [Watson et al., 1985] Watson, L., Laffey, T., e Haralick, R. (1985). Topographic classification of digital image intensity surfaces using generalized splines and the discrete cosine transform. *Computer Vision, Graphics, and Image Processing*, 29(2):143–167.
- [Whitley, 1989] Whitley, D. (1989). The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best. In *Proceedings of the International Conference on Genetic Algorithms*, páginas 116–121.
- [Yokoya e Levine, 1989] Yokoya, N. e Levine, M. (1989). Range image segmentation based on differential geometry: A hybrid approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6):643–649.
- [Yu et al., 1994] Yu, X., Bui, T., e Krzyzak, A. (1994). Robust estimation for range image segmentation and reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):530–538.
- [Zhang e Kim, 2000] Zhang, B. e Kim, J. (2000). Comparison of selection methods for evolutionary optimization. *Evolutionary Optimization*, 2(1):55–70.